



Titre: Modélisation géométrique d'ailes d'avion pour l'optimisation
Title: multidisciplinaire

Auteur: Sylvain Marier
Author:

Date: 2005

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Marier, S. (2005). Modélisation géométrique d'ailes d'avion pour l'optimisation
Citation: multidisciplinaire [Mémoire de maîtrise, École Polytechnique de Montréal].
PolyPublie. <https://publications.polymtl.ca/7640/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/7640/>
PolyPublie URL:

**Directeurs de
recherche:**
Advisors:

Programme: Non spécifié
Program:

UNIVERSITÉ DE MONTRÉAL

MODÉLISATION GÉOMÉTRIQUE D'AILES D'AVION POUR L'OPTIMISATION
MULTIDISCIPLINAIRE

SYLVAIN MARIER
DÉPARTEMENT DE GÉNIE MÉCANIQUE
ÉCOLE POLYTECHNIQUE DE MONTRÉAL

MÉMOIRE PRÉSENTÉ EN VUE DE L'OBTENTION
DU DIPLÔME DE MAÎTRISE ÈS SCIENCES APPLIQUÉES
(GÉNIE MÉCANIQUE)
DÉCEMBRE 2005



Library and
Archives Canada

Bibliothèque et
Archives Canada

Published Heritage
Branch

Direction du
Patrimoine de l'édition

395 Wellington Street
Ottawa ON K1A 0N4
Canada

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

ISBN: 978-0-494-16812-7

Our file Notre référence

ISBN: 978-0-494-16812-7

NOTICE:

The author has granted a non-exclusive license allowing Library and Archives Canada to reproduce, publish, archive, preserve, conserve, communicate to the public by telecommunication or on the Internet, loan, distribute and sell theses worldwide, for commercial or non-commercial purposes, in microform, paper, electronic and/or any other formats.

The author retains copyright ownership and moral rights in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

AVIS:

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque et Archives Canada de reproduire, publier, archiver, sauvegarder, conserver, transmettre au public par télécommunication ou par l'Internet, prêter, distribuer et vendre des thèses partout dans le monde, à des fins commerciales ou autres, sur support microforme, papier, électronique et/ou autres formats.

L'auteur conserve la propriété du droit d'auteur et des droits moraux qui protègent cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

In compliance with the Canadian Privacy Act some supporting forms may have been removed from this thesis.

Conformément à la loi canadienne sur la protection de la vie privée, quelques formulaires secondaires ont été enlevés de cette thèse.

While these forms may be included in the document page count, their removal does not represent any loss of content from the thesis.

Bien que ces formulaires aient inclus dans la pagination, il n'y aura aucun contenu manquant.


Canada

UNIVERSITÉ DE MONTRÉAL

ÉCOLE POLYTECHNIQUE DE MONTRÉAL

Ce mémoire intitulé:

MODÉLISATION GÉOMÉTRIQUE D'AILES D'AVION POUR L'OPTIMISATION
MULTIDISCIPLINAIRE

présenté par: MARIER Sylvain

en vue de l'obtention du diplôme de: Maîtrise ès sciences appliquées

a été présentée au jury d'examen constitué de:

M. PELLETIER Dominique, Ph.D., président

M. TRÉPANIÉ Jean-Yves, Ph.D., membre et directeur de recherche

M. GUIBAULT François, Ph.D., membre et codirecteur de recherche

M. RIVEST Louis, Ph.D., membre

Science sans conscience n'est que ruine de l'âme

-Rabelais, 1532

REMERCIEMENTS

Je voudrais remercier dans un premier temps mes directeurs de recherche, Jean-Yves Trépanier et François Guibault. M. Trépanier m'a immédiatement fait confiance en me proposant ce projet, une composante d'un projet de grande envergure, dont il est le directeur. Pour cela, tout comme pour les conseils avisés et sa disponibilité, je lui suis très reconnaissant. M. Guibault m'a transmis un savoir au niveau de la manipulation des courbes paramétriques et d'objets informatiques.

Ensuite, je mentionne la grande disponibilité et les discussions enrichissantes faites avec Christophe Tribes, un associé de recherche de M. Trépanier. Ses qualités d'écoute et son expertise en optimisation ont été de grande aide lorsque des difficultés se sont présentées en chemin.

L'implication de l'entreprise dans le projet, Bombardier Aéronautique dans mon cas, est un facteur de motivation. Je remercie donc François M. Pépin et David Leblond, du département d'aérodynamique avancée, pour leur collaboration au projet. L'aide de M. Pépin pour définir les besoins et requis a été essentielle et M. Leblond nous a fourni des géométries.

Ensuite, je souligne la contribution de tous les collègues du bureau pour le support et les sources de conseils et d'idées inépuisables qui rend la vie au local agréable. Particulièrement, je remercie Jérôme Bettinger, qui a aussi été mon binôme dans plusieurs cours, dont l'expérience et la débrouillardise sur différentes plates-formes informatiques est un atout indéniable. Notons aussi l'appui d'Alexandre, d'Amadou, de Pascal, de tous les stagiaires qui sont passés, et des nouveaux arrivés, pour la motivation contagieuse qui se dégage de cette collaboration.

Enfin, les dernières personnes que je voudrais remercier, mais non les moindre, sont mes proches et amis. Sans le support apporté par ceux-ci, j'aurais eu bien des difficultés à passer à travers ces années. Un merci tout particulier à ma famille, Pierre, Lucie, Johanne et Valérie et à ma copine Amélie.

RÉSUMÉ

Ce mémoire traite du développement d'une paramétrisation pour la représentation de la surface extérieure d'une aile d'avion civil volant en régime transsonique. Un programme informatique, devant s'intégrer à une architecture de systèmes permettant de faire des optimisations multidisciplinaires, s'appuyant sur cette paramétrisation a été développé.

La paramétrisation développée s'appuie sur des grandeurs physiques, appelées *paramètres d'ingénierie*, permettant ainsi à l'ingénieur expérimenté de bien saisir la nature des paramètres gouvernant la génération de la surface de l'aile. Ces paramètres affectent la définition de la forme en plan et des profils guides.

Deux paramétrisations ont été développées, s'appuyant toutes deux sur le même principe : l'aile est définie à l'aide de trois profils guides situés à des endroits stratégiques, soient l'emplanture, la cassure et le bout de l'aile. La définition de ces profils s'appuie, quant à elle, sur les paramètres d'ingénierie. Ces trois profils guides servent de base à la génération d'un squelette de caractéristiques géométriques dans le sens de l'envergure. Sur ce dernier, la définition d'un profil situé à une station quelconque de l'aile peut être extraite.

Les paramètres d'ingénierie gouvernant la génération des profils sont regroupés en plusieurs couches, chacune gouvernant une caractéristique de l'aile (longerons, bec & volet, ...), celles-ci pouvant être activées ou non à la guise de l'utilisateur. Ceci permet de contrôler le nombre de paramètres définissant la surface de l'aile en fonction du contrôle voulu et du niveau de détail désiré.

Afin de démontrer l'étendue et les limites des représentations possibles, plusieurs profils courants et une aile typique ont été représentés à l'aide de cette paramétrisation.

ABSTRACT

This master thesis describes a new parameterization for the representation of the external surface of an airplane wing in transonic flight condition. A software using this parameterization which can be integrated to a multidisciplinary optimization framework as been developed.

The new parameterization is based on physical quantities, called *engineering parameters*, allowing an experienced engineer to understand the nature of the parameters governing the wing skin generation process. These parameters are used for the definition of the planform and the master stations.

Two parameterizations have been developed, both using the same principle : the wing is defined with three masters stations located at strategic chordwise locations : the root, the break and the tip of the wing. The geometric definition of these stations is based on the engineering parameters. These three master stations serve as the basis for the generation of a constraints skeleton in the spanwise direction. With this skeleton, the geometric definition of a station situated at any location of the wing can be obtained.

The engineering parameters governing the stations generation are grouped together in layers in a hierarchical way allowing them to be activated or not by the user. This allows controlling the number of parameters defining the surface of the wing according to the desired control and level of detail.

In order to show the range and the limits of possible representations, several usual airfoils and a reference wing have been represented with this parameterization.

TABLE DES MATIÈRES

DÉDICACE	iv
REMERCIEMENTS	v
RÉSUMÉ	vi
ABSTRACT	vii
TABLE DES MATIÈRES	viii
LISTE DES TABLEAUX	xii
LISTE DES FIGURES	xiii
LISTE DES NOTATIONS ET DES SYMBOLES	xv
LISTE DES ANNEXES	xviii
INTRODUCTION	1
0.1 Situer le projet dans son environnement de recherche	2
0.2 Objet du mémoire	4
0.3 Organisation du document	5
CHAPITRE 1 CONCEPTS PRÉLIMINAIRES	7
1.1 Revue des méthodes de paramétrisation géométrique pour le design aérodynamique	7
1.1.1 Transformations conformes	8
1.1.2 Les profils NACA	12
1.1.3 Approche discrète	16

1.1.4	Paramétrisation PARSEC	17
1.1.5	Bases polynomiales pour la représentation de courbe	23
1.1.6	Courbes de Bézier	25
1.1.7	Courbes B-Splines & NURBS	28
1.1.8	Fonctions de formes et fonctions de bases orthogonales	30
1.1.9	Équations aux dérivées partielles	31
1.1.10	Outils de CAO	32
1.1.11	Approche orienté sur les déformations libres	33
1.2	Critères de Sélection	34
CHAPITRE 2	DESCRIPTION DES BESOINS	39
2.1	Généralités	39
2.2	Profils 2D	41
2.2.1	Rappel des caractéristiques des profils	41
2.2.2	Les paramètres des profils 2D	43
2.3	Les paramètres de la forme en plan	47
2.4	Les caractéristiques de l'aile 3D	49
2.5	Logiciel	51
CHAPITRE 3	COURBES CONTRAINTES	54
3.1	B-Splines contraintes	54
3.1.1	B-Spline contrainte aux extrémités	56
3.1.1.1	De contraintes à dérivées	59
3.1.1.2	Méthode de construction du vecteur nodal	63
3.1.1.3	Collage final des courbes	68
3.1.1.4	Diminution de l'ordre des points clés	69
3.1.2	B-Spline multi-contraintes en plusieurs endroits	71
3.1.2.1	Construction du vecteur nodal	74
3.1.2.2	Construction du membre de droite	79

3.1.2.3	Remplissage de la matrice du système	80
3.1.3	Optimisation des amplitudes des dérivées premières	82
3.1.4	Comparaison des méthodes	83
3.2	Interpolation conique dans le sens de l'envergure	89
CHAPITRE 4	CONSTRUCTION DE L'AILE	95
4.1	Premier module avec la définition PARSEC	95
4.2	Deuxième module utilisant des B-Splines	98
4.2.1	Contraintes multi-couches	100
4.2.2	Construction des stations guides	102
4.2.3	Construction des bec et volet	104
4.2.4	Construction des surfaces intérieures	106
4.2.5	Discrétisation	108
CHAPITRE 5	VALIDATION GÉOMÉTRIQUE	112
5.1	Formulation	112
5.2	Validation géométrique sur des profils 2D	113
5.2.1	Mise en place du problème d'optimisation bidimensionnel . . .	114
5.2.2	Comparaison des deux méthodes de génération des B-splines .	117
5.2.3	Comparaison des optimisations avec et sans amplitude	121
5.2.4	Effet du nombre de couches	125
5.3	Validation géométrique sur l'aile typique BAW	132
5.3.1	Mise en place du problème d'optimisation tridimensionnel . . .	132
5.3.2	Effet de la fonction coût	135
5.3.3	Effet du nombre de couches	138
5.3.4	Comparaison des variations des caractéristiques géométriques de l'aile et des coniques servant à l'interpolation dans le sens de l'envergure	140

CONCLUSION	147
6.1 Contribution en fonction des objectifs énumérés	148
6.2 Discussions	149
6.3 Travaux futurs	151
ANNEXES	157

LISTE DES TABLEAUX

TAB. 1	Liste des différentes tâches du projet MOSAIC	4
TAB. 1.1	Tableau synthèse des différentes approches	37
TAB. 3.1	Construction du vecteur nodal	75
TAB. 4.1	Tableau résumé des couches de contraintes	102
TAB. 5.1	Points clés pour la validation 2D	116
TAB. 5.2	Paramètres de la forme en plan de l'aile BAW	133
TAB. 5.3	Caractéristiques géométriques de l'aile BAW	134
TAB. 5.4	Résultats obtenus lors des optimisations géométriques de l'aile BAW à l'aide des deux différentes fonctions coûts	137
TAB. 5.5	Détails des distances et des erreurs sur les stations de l'aile après les optimisations	139
TAB. III.1	Nature du fichier de construction	173
TAB. III.2	Nature du fichier des paramètres de la forme en plan	176
TAB. III.3	Nature du fichier des paramètres géométriques des profils guides	178
TAB. III.4	Nature du fichier des paramètres géométriques des profils guides (suite)	179

LISTE DES FIGURES

FIG. 1.1	Transformation de Joukowski	10
FIG. 1.2	Transformation de Joukowski étendue	11
FIG. 1.3	Profil NACA 0012 et 2412	13
FIG. 1.4	Paramètres PARSEC	18
FIG. 1.5	Profil avec la paramétrisation PARSEC	23
FIG. 1.6	Courbe polynomiale	25
FIG. 1.7	Courbe de Bézier	27
FIG. 2.1	Profil conventionnel vs supercritique	42
FIG. 2.2	Paramètres géométriques d'un profil 2D	44
FIG. 2.3	Paramètres géométriques de la forme en plan	48
FIG. 2.4	Distribution de torsion	51
FIG. 3.1	Schéma d'imposition des contraintes aux extrémités	56
FIG. 3.2	Effet de l'amplitude de la dérivée sur une courbe	61
FIG. 3.3	Effet des différentes méthodes de construction du vecteur nodal	64
FIG. 3.4	Longueur de courbe en fonction de l'avancement paramétrique	68
FIG. 3.5	Collage final de B-Splines primaires	70
FIG. 3.6	Fonction de modification de la distance entre les nœuds	77
FIG. 3.7	Profil approximé selon la méthode PARSEC	84
FIG. 3.8	Profil approximé selon la première méthode	87
FIG. 3.9	Profil approximé selon la deuxième méthode	88
FIG. 3.10	Méthode de construction d'une conique	90
FIG. 3.11	Exemples de coniques interpolant trois points	94
FIG. 4.1	Problèmes liés à la rotation d'un profil	97
FIG. 4.2	Seconde approche pour la rotation d'un profil	98
FIG. 4.3	Aile construite avec le premier module	99
FIG. 4.4	Couches de contraintes	100

FIG. 4.5	Construction de l'aile, étape 1	103
FIG. 4.6	Construction de l'aile, étape 2	105
FIG. 4.7	Construction de l'aile, étape 3	107
FIG. 4.8	Effet du paramètre de décalage	110
FIG. 4.9	Exemple de discrétisation de profils	111
FIG. 4.10	Construction de l'aile, étape 4	111
FIG. 5.1	Effet du pas de différenciation sur la fonction coût	118
FIG. 5.2	Profil construit à l'aide de la méthode #2	119
FIG. 5.3	Comparaison des écarts maximum	122
FIG. 5.4	Design initial pour la validation géométrique	123
FIG. 5.5	Design final avec amplitude et avec tous les paramètres	124
FIG. 5.6	Comparaison des optimisations avec et sans amplitude	126
FIG. 5.7	Écart maximum en fonction du nombre de couches	127
FIG. 5.8	Écart maximum en fonction des couches activées, par profil	128
FIG. 5.9	Optimisations avec un nouveau point de départ	130
FIG. 5.10	Résultats d'optimisation géométrique de différents profils	131
FIG. 5.11	Comparaison des deux fonctions coûts sur l'aile	138
FIG. 5.12	Fonction coût en fonction des couches actives	140
FIG. 5.13	Erreur de représentation pour le design a	141
FIG. 5.14	Erreur de représentation pour le design d	142
FIG. 5.15	Meilleur aile et variation des paramètres selon l'envergure	143
FIG. 5.16	Variation des longerons dans le sens de l'envergure	144
FIG. 5.17	Variation des autres paramètres dans le sens de l'envergure	145

LISTE DES NOTATIONS ET DES SYMBOLES

AA	Aérodynamique Avancée (<i>Advanced Aerodynamics</i>)
AG	Algorithme Génétique
BA	Bombardier Aéronautique (<i>Bombardier Aeronautic</i>)
CAD	<i>Computer Aided Design</i>
CAO	Conception Assistée par Ordinateur
CFD	<i>Computational Fluid Dynamic</i>
CRIAQ	Consortium de Recherche et d'Innovation en Aérospatiale au Québec
EDP	Équations aux dérivées partielles
FFD	<i>Free Form Deformation</i>
MDO	<i>Multidisciplinary Design & Optimization</i>
MOSAIC	<i>Multidisciplinary Optimization Standardization Approach for Integration and Configurability</i>
NACA	<i>National Advisory Committee for Aeronautics</i>
NASA	<i>National Aeronautics and Space Administration</i>
NURBS	<i>Non-Uniform Rational B-Spline</i>
LE	<i>Leading Edge</i> (Bord d'attaque)
TE	<i>Trailing Edge</i> (Bord de fuite)
IN	Point minimum situé sur l'intrados
EX	Point maximum situé sur l'extrados
RSI	<i>Rear Spar Intrados</i> (Longeron arrière sur l'intrados)
FSI	<i>Front Spar Intrados</i> (Longeron avant sur l'intrados)
FSE	<i>Front Spar Extrados</i> (Longeron avant sur l'extrados)
RSE	<i>Rear Spar Extrados</i> (Longeron arrière sur l'extrados)
FI	<i>Flap Intrados</i> (Volet intrados)
SI	<i>Slat Intrados</i> (Bec intrados)
SE	<i>Slat Extrados</i> (Bec extrados)
FE	<i>Flap Extrados</i> (Volet extrados)
Cav	Cavité
Inf	Inflexion
SSI	<i>Supplementary Spar Intrados</i> (Longeron auxiliaire sur l'intrados)
SSE	<i>Supplementary Spar Extrados</i> (Longeron auxiliaire sur l'extrados)

Caractères usuels

c	Corde
c_r	Corde réelle
c_x	Corde projetée sur x
c_l	Coefficient de portance
c_p	Coefficient de pression
$\mathbf{C}(u)$	Courbe paramétrique
$f(\mathbf{x})$	Fonction coût ou objective
$[\mathbf{H}]$	Matrice Hessienne
l	Nombre de points clés à interpoler
L	Longueur de corde d'une courbe
m	Nombre de nœuds dans le vecteur nodal
M	Nombre de Mach
n	Nombre de points de contrôle de la courbe
n_{ni}	Nombre de nœuds internes dans le vecteur nodal
N	Nombre de contraintes à imposer
$N_{i,p}(u)$	$i^{\text{ème}}$ fonction de base b-splinaire de degré p
p	Degré de la courbe
\mathbf{P}_i	$i^{\text{ème}}$ point de contrôle
\mathbf{Q}_i^f	$i^{\text{ème}}$ point clé d'ordre f
u_i	$i^{\text{ème}}$ élément du vecteur nodal
u_r	Paramètre d'avancement sur la courbe
\bar{u}_i	Paramètre d'avancement associé au $i^{\text{ème}}$ point clé
\mathbf{U}	Vecteur nodal

Caractères grecs

α	Angle d'incidence du profil
θ	Angle du bord de fuite
ϕ	Angle de coin du bord de fuite
γ	Direction de i à $i + 1$
Λ	Angle de flèche de l'aile

∇	Opérateur gradient
ρ, R	Rayon de courbure
κ	Courbure

Indices

r	Root (<i>Emplanture</i>)
b	Break (<i>Cassure</i>)
t	Tip (<i>Bout</i>)
in	intrados
ex	extrados
ni	Nœuds internes
x	Composante <i>X</i>
y	Composante <i>Y</i>
U	Borne supérieure (<i>Upper</i>)
L	Borne inférieure (<i>Lower</i>)
\angle	Angle
\leftrightarrow	Amplitude

Symboles

X	Scalaire
\mathbf{X}	Vecteur
\mathbf{X}^T	Vecteur transposé
$[\mathbf{X}]$	Matrice
x'	Dérivée première
x''	Dérivée seconde

LISTE DES ANNEXES

ANNEXE I	RAPPEL : COURBES B-SPLINAIRES	157
I.1	Fonctions de bases	157
I.2	Courbes B-Splinaires	159
I.3	Dérivées des courbes B-Splinaires	160
I.4	Manipulations contraintes de courbes B-Splinaires	162
ANNEXE II	RAPPEL : MÉTHODES D'OPTIMISATION À GRADIENT QUASI- NEWTON BFGS	167
II.1	Formulation	167
II.2	Optimiseur à base de gradient quasi-Newton BFGS	168
ANNEXE III	DESCRIPTION DES FICHIERS D'ENTRÉES	172
III.1	Paramètres généraux de construction de l'aile	172
III.2	Paramètres de la forme en plan de l'aile	175
III.3	Paramètres géométriques de l'aile	177

INTRODUCTION

AVEC la mondialisation de l'économie, les pressions exercées sur les entreprises pour demeurer concurrentielles sont énormes. Ces pressions proviennent autant des demandes exigeantes des clients (coûts, livraison, etc.) que de la concurrence exercée par les autres joueurs de l'industrie ou encore de décisions politiques. Il est donc primordial de trouver des moyens efficaces pour conserver et gagner de nouvelles parts de marché.

Dans les industries où les phases de conception, fabrication et service à la clientèle sont réparties à travers le monde, comme celle de l'automobile et de l'aéronautique, ces pressions sont encore plus grandes. Dans ces cas, les décisions de nature économiques et géopolitiques ont de grandes influences sur la performance d'une entreprise. Mais au-delà de cette donne, la capacité d'une entreprise d'être concurrentielle dépend de la valeur du produit. Pour l'industrie des constructeurs d'avion civils, la valeur se mesure autant en coût de vente et d'entretien qu'en délai de conception, de production et de livraison.

Comme les coûts d'un appareil peuvent être évalués relativement tôt dans le processus de design, la rapidité d'un constructeur à développer et certifier un nouvel appareil est un facteur déterminant sur la place qu'occupera ce dernier dans le marché mondial. Quand un constructeur juge qu'il doit investir pour lancer un nouvel avion afin d'occuper une niche de marché, plusieurs stratégies peuvent être mises de l'avant pour diminuer le temps de design. Par exemple, les constructeurs aéronautiques développent, entre autre, des versions modifiées d'appareils existants. Bombardier Aéronautique a déjà fait usage de cette stratégie, et continue toujours de l'utiliser, tout comme ses concurrents.

Une autre approche pour aider à réduire le cycle de design est de développer une architecture de design et d'optimisation multidisciplinaires automatisée. Pour mener à bien de telles optimisations, une multitude de modules doivent être développés afin de lire et

convertir des informations, lancer des calculs, extraire des résultats et procéder à différentes analyses. L'optimiseur maître coordonne les interactions entre ces différents modules. C'est également celui-ci qui fait le travail pour trouver les meilleurs compromis en minimisant la fonction objective à partir de designs initiaux.

Bombardier Aéronautique, et en particulier le département d'Aérodynamique Avancée (AA), a manifesté son intérêt dans le développement et l'utilisation de nouveaux outils adaptés à l'optimisation multidisciplinaire. Cet intérêt s'est concrétisé par le développement d'algorithmes et d'approches pour solutionner des problèmes d'optimisations, dont quelques-uns développés avec la collaboration de l'École Polytechnique de Montréal.

Comme cette compagnie est subdivisée en départements qui ont développés avec le temps leurs expertises respectives, les échanges d'informations des uns aux autres deviennent plus complexe. En effet, il arrive que les différents départements n'utilisent pas tous les mêmes méthodes, les mêmes modèles ou les mêmes formats. Cette situation se présente dans la majorité des grandes entreprises, où le produit fini est complexe. Dans ces conditions, il est plutôt difficile de faire rapidement de bons designs, vu la quantité de gens et de logiciels, donc de méthodes de travail, de formats et de modèles informatiques, qui sont impliqués dans la boucle de conception.

0.1 Situer le projet dans son environnement de recherche

C'est dans l'objectif de simplifier ces tâches qu'est né le projet MOSAIC. Il a comme objectif de permettre le design et l'optimisation de composantes et/ou systèmes en tenant compte de résultats d'analyses provenant de une ou plusieurs discipline(s) (optimisation multidisciplinaire — MDO). Il implique trois universités (École Polytechnique, École de Technologie Supérieure (ÉTS) et Concordia) et fait intervenir les trois grandes compagnies aéronautiques du Québec (Bombardier Aéronautique, Pratt & Whitney et Bell

Hélicoptère). Ce projet est financé par ces partenaires industriels ainsi que par le gouvernement québécois à travers le CRIAQ¹. Le projet MOSAIC vise à explorer diverses composantes des disciplines touchant l'optimisation de systèmes aéronautiques dans le but de développer de l'expertise en MDO chez les industries participantes. Ce projet s'est fixé les trois principaux objectifs suivants :

1. Le développement d'un protocole d'échange de donnée permettant une bonne intégration des analyses et des logiciels d'optimisations dans un contexte de MDO,
2. Le développement de logiciels d'analyses et d'optimisations spécialisés adaptés pour la MDO,
3. L'intégration de ces logiciels dans une architecture d'optimisation configurable pour les besoins des différentes industries.

Pour mener à terme ce projet, une structure a été mise en place et le travail décomposé en plusieurs tâches. Des tâches de nature plus générique vont toucher tous les acteurs du projet, alors que les tâches plus spécialisées verront leur étendue davantage limitée. Le tableau 1 montre cette division des tâches. Notons que les cinq premières sont génériques.

Le projet MOSAIC implique donc plusieurs des grandes disciplines relatives à l'aérospatiale. Une d'entre elles est la modélisation paramétrique de formes géométriques. C'est justement sur celle-ci que va se concentrer le présent mémoire (tâche 8 du tableau 1) : développer un module de génération d'aile d'avion adapté à l'optimisation multidisciplinaire. L'objectif est de fournir aux ingénieurs aéronautiques un outil précis et flexible permettant la manipulation des formes extérieures pendant le processus de design. Comme ce module est développé pour Bombardier, il devra éventuellement s'insérer de façon harmonieuse parmi les outils de représentation des surfaces d'aile de la compagnie.

¹www.criaq.aero et www.criaq.aero/Francais/Projets/4_1.html

TAB. 1: Liste des différentes tâches du projet MOSAIC

Numéro de la tâche	Nom descriptif
1	Protocole d'échange de données
2	Optimisation
3	Décomposition du problème
4	Design en environnement immersif
5	Modèle digital structuré
6	Optimisation structurelle
7	Charges
8	Module de génération d'aile
9	Aéroélasticité
10	MDO préliminaire pour les turbines
11	Module de génération d'aubage de turbine

0.2 Objet du mémoire

Le présent travail comprend trois volets. Premièrement, il y a l'élaboration d'une nouvelle paramétrisation adaptée aux ailes transsoniques avec cassures et angle de flèche. Pour arriver à développer cette paramétrisation, nous devons d'abord identifier les paramètres clés sur les profils adimensionnels, la forme en plan et l'aile complète afin de permettre une représentation flexible. Pour des fins pratiques, ces paramètres du modèle géométrique devront avoir une signification physique pour l'ingénieur d'expérience. Aussi, l'utilisateur doit pouvoir contrôler le nombre de paramètres en fonction de l'application, de la précision désirée et du temps de calcul disponible. Le modèle géométrique doit prendre en compte une série de contraintes permettant la modélisation d'ailes réalistes et manufacturables ainsi que l'amélioration d'ailes existantes. De plus, le modèle géométrique doit rester le plus générique possible, dans le but d'explorer de multiples configurations.

Le deuxième volet concerne la création d'un programme informatique de construction de la surface d'une aile se basant sur cette paramétrisation. La génération de l'aile à

partir des paramètres devra être totalement automatique dans l'optique de garder l'intégration du module dans une boucle d'optimisation la plus simple possible. Aussi, les fichiers d'entrées tous comme ceux de sorties produits par le module de génération de la surface doivent être facilement utilisables et modifiables. Il faut garder en tête que le module informatique doit pouvoir communiquer aisément, via un protocole d'échange de donnée, avec la maquette numérique centrale (respectivement, les tâches 1 et 5 du tableau 1). De plus, des formats adaptés aux applications spécifiques (maillage et calcul aérodynamique) sont à prévoir autant qu'un transfert vers un format standard.

Enfin, vient une partie validation, tant au niveau bidimensionnel que tridimensionnel. Il faut s'assurer que la nouvelle paramétrisation et le module de génération puissent représenter de façon adéquate plusieurs profils et ailes existantes. Pour ce faire, nous ferons une optimisation géométrique à l'aide de la méthode quasi-Newton BFGS. La fonction coût utilisée sera alors la distance entre un profil ou une aile cible et le résultat de la génération.

0.3 Organisation du document

Ce mémoire résume le travail qui a été fait pour atteindre les objectifs fixés. Les principales étapes nécessaires pour arriver à un module fonctionnel et l'optimisation qui s'en suit sont présentées dans l'ordre logique de développement. Ainsi, le chapitre 1 commence par une brève revue de littérature de différents types de représentation de courbes et surfaces adaptées à l'aéronautique. Une méthode d'évaluation sert à nous guider dans le choix du type de représentation retenu. Par la suite, au chapitre 2, une description en détail des différents besoins et requis que doit rencontrer le module est présentée. Le chapitre 3 présente des techniques de constructions des différents types de courbes qui se trouvent dans le modèle. La première partie vise l'interpolation des points clés pour obtenir un profil et la deuxième traite de l'interpolation dans le sens de l'envergure.

Le chapitre 4 explique le processus de construction d'aile. Le squelette de caractéristiques géométriques y est vu pour les deux approches testées. Enfin, au chapitre 5, il est question de l'optimisation géométrique proprement dite. Dans un premier temps, l'expérience est tentée sur un profil 2D puis sur l'aile entière. Un retour sur le travail effectué et les possibilités offertes par ce module sont présentés en conclusion. En annexe sont présentés certains renseignements supplémentaires relatifs à l'implémentation informatique du module. Ainsi, un rappel mathématique sur les B-Splines est présenté à l'annexe I. Un autre sur la méthode d'optimisation utilisée pour cette recherche, la méthode à gradient quasi-Newton BFGS, est présenté à l'annexe II. Enfin, l'annexe III, décrit les fichiers d'entrées du module.

CHAPITRE 1

CONCEPTS PRÉLIMINAIRES

AFIN de pouvoir étudier notre problème d'un regard éclairé, nous présentons dans un premier temps une brève revue de littérature. Celle-ci survole d'abord différentes méthodes de paramétrisation utilisées pour représenter des profils et des ailes. Une attention particulière est mise sur deux d'entre elles - PARSEC (section 1.1.4) et B-Spline (section 1.1.7 et annexe I) - car elles ont été retenues pour comparaison dans la présente étude. Une énumération des critères de sélection pour le choix de représentation des courbes est présentée afin de comparer les différentes méthodes. Un tableau de synthèse et une discussion sur les choix de courbes complètent cette énumération.

1.1 Revue des méthodes de paramétrisation géométrique pour le design aérodynamique

Cette section fait un bref survol des techniques de paramétrisation et tente de les classer de manière générale. Un classement inspiré de celui de Samareh (1999) est présenté. Les techniques énumérées ici ont été sélectionnées dû à leur caractère historique ou pour la place qu'elles occupent aujourd'hui dans l'industrie aéronautique.

Plusieurs techniques de paramétrisation ont été développées au cours des années pour représenter, majoritairement des profils, et quelques fois, des ailes entières. Ces techniques ont toutes leurs avantages, mais aussi leurs limitations. Le choix d'une méthode de paramétrisation à une grande incidence sur les capacités de représentation, donc sur l'étendue des designs possibles.

Méthodes de paramétrisation 2D

Nous nous concentrerons davantage dans un premier temps sur les méthodes servant à la paramétrisation de profils aérodynamiques 2D. L'approche 2D a souvent été privilégiée dans le passé. C'est une façon de faire très intuitive qui donne une bonne approximation des caractéristiques de l'aile à partir de celles du profil. Aussi, lorsque nous avons une bonne paramétrisation 2D, il est possible de l'étendre en 3D par l'ajout d'une dimension.

1.1.1 Transformations conformes

Une des premières méthodes employée pour paramétrer des formes aérodynamique est l'utilisation des transformations conformes (Glauert 1947, Theodorsen et Garrick 1934). Celles-ci font la transposition d'un espace imaginaire $z = x + iy$ ($x, y \in \mathbb{R}$, $i = \sqrt{-1}$ et $z \in \mathbb{C}$), vers un deuxième $\zeta = \xi + i\eta$. Cette projection bijective est obtenue à partir d'une transformation $\zeta = f(z)$. Les transformations conformes ont une place importante en analyse complexe et dans plusieurs sphères de la physique et de l'ingénierie.

Certaines de ces transformations ont trouvé une place en aéronautique, car elles font apparaître naturellement des formes qui ressemblent à certaines composantes aérodynamiques. Notons la transformation $\zeta = e^z + z$, qui transforme les droites $y = x$ en courbes rappelant la forme des prises d'air de réacteurs. Ou encore, la transformation de Joukowski qui change un cercle en une forme se rapprochant d'un profil aérodynamique.

L'utilité de transformer un cercle en un profil est davantage historique qu'actuelle. Avant l'ère du transistor et des simulations numériques, les solutions analytiques d'écoulements simples étaient recherchées pour valider certaines expériences. On détenait la solution analytique de plusieurs écoulements potentiels autour de corps simples, tel des plaques et des cercles. En ayant le moyen de transformer les cercles en profils, ce que

fournit la transformation de Joukowski, il devient possible de déduire l'écoulement autour de corps plus complexes. En appliquant cette transformation, on obtient la géométrie du profil et l'écoulement potentiel autour de celui-ci et donc, les caractéristiques aérodynamiques du profil. C'est là que réside l'utilité de cette transformation : la possibilité d'estimer les performances d'un profil rapidement et sans recourir aux tests expérimentaux.

La transformation de Joukowski s'exprime comme suit :

$$\zeta = z + \frac{1}{z} \quad (1.1)$$

Avec $z = x + iy$, un nombre complexe. Il n'y a que deux paramètres contrôlant cette transformation. Il s'agit des parties réelle et imaginaire de la position du centre du cercle (x_c, y_c) à transformer. On sait que x_c agit sur l'épaisseur du profil et que y_c influence sa cambrure. La figure 1.1 montre un profil avec $x_c = y_c = .05$ et illustre les effets de ces paramètres.

La transformation précédente ne permet pas beaucoup de liberté avec seulement deux variables de design. En faisant une première transformation sur z , la transformation gagne en flexibilité. C'est ce que fait la transformation étendue de Joukowski. Elle s'exprime comme suit :

$$\zeta = z' + \frac{1}{z'} \quad (1.2)$$

Avec comme première transformation sur z :

$$z' = z - \frac{\varepsilon}{z - \Delta} \quad (1.3)$$

Où ε est un nombre complexe et Δ un réel. De cette façon, il y a cinq paramètres pour définir le profil :

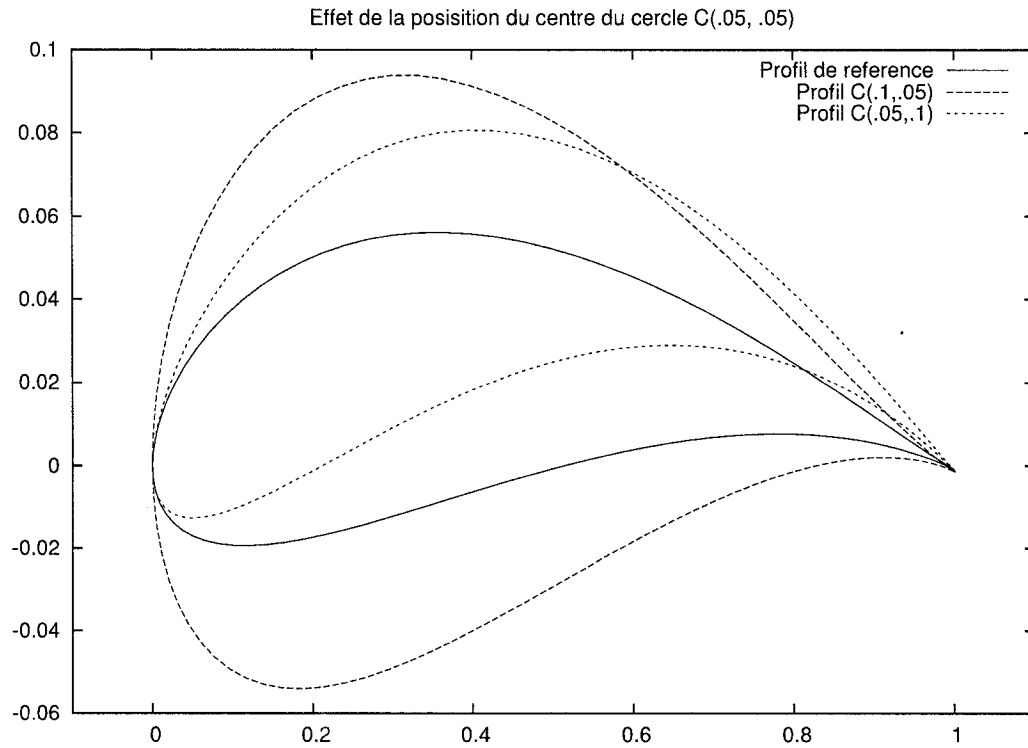


FIG. 1.1: Profils résultant de la transformation de Joukowski

x_c, y_c les parties réelle et imaginaire du centre du cercle ;

x_t, y_t les parties réelle et imaginaire de ε ;

Δ une première translation suivant l'axe des x .

Bien qu'en général x_c et x_t influencent l'épaisseur et y_c et y_t la cambrure, certains effets couplés semblent parfois apparaître. Ainsi, il est facilement possible de créer des formes ne ressemblant plus à des profils. Ce phénomène rend les manipulations du profil difficiles avec cette transformation. La figure 1.2 représente un profil obtenu avec la transformation étendue sur lequel l'effet des paramètres x_t et y_t est illustré.

Pour avoir encore plus de flexibilité, Theodorsen et Garrick (1934) proposent une nouvelle transformation. Celle-ci peut changer un cercle en n'importe quel forme, donc tous les profils peuvent être représentés. Ici encore il s'agit de faire une première transforma-

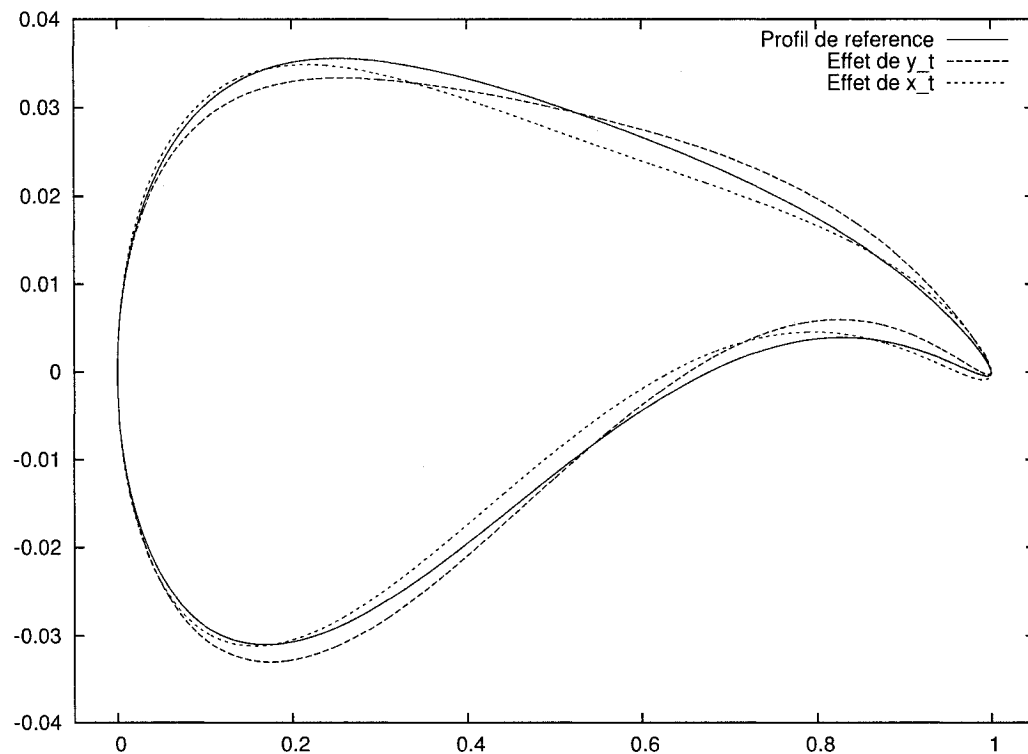


FIG. 1.2: Profils résultants de la transformation de Joukowski étendue

tion sur z avant de faire la transformation de Joukowski. Comme auparavant, $\zeta = z' + \frac{1}{z'}$, mais avec comme première transformation sur z :

$$z' = z \exp \left(\sum_{n=0}^{\infty} \frac{C_n}{z^n} \right) \quad (1.4)$$

Une série infinie est requise pour faire cette transformation, ce qui est impossible informatiquement. Dans la pratique, une série tronquée plus ou moins longue selon le degré de précision voulu est utilisée. Cependant, les coefficients dans cette série ne représentent rien de tangible, rendant leurs interprétations extrêmement difficiles. Comme les possibilités de la transformation sont infinies, il n'est pas aisé d'obtenir une courbe résultante ressemblant à un profil. Aussi, il est difficile de prévoir l'effet d'une troncature sur la courbe après le $n^{\text{ième}}$ terme.

Ces techniques ne sont plus utilisées fréquemment aujourd'hui. Bien qu'elles puissent parvenir à bien représenter certains profils, elles n'en demeurent pas moins plutôt limitées. Elles ont comme principaux inconvénients de ne pas offrir de corrélation physique entre les paramètres et la géométrie. Bien que ce ne soit pas nécessairement un problème pour un optimiseur, ça peut en être un pour l'ingénieur aéronautique. Elles ne supportent pas non plus le contrôle local. Ceci est un grand problème lorsqu'il faut changer une partie seulement d'un modèle géométrique.

1.1.2 Les profils NACA

Le NACA a grandement fait avancer les recherches en terme de paramétrisation de profils. L'organisme, aujourd'hui connu sous l'acronyme NASA, a publié plusieurs séries de famille de profils. Ces familles de sections sont encore des références documentées de nombreux résultats expérimentaux. Nous nous attarderons principalement sur la famille à quatre chiffres.

NACA 4-digit

Cette famille est caractérisée par trois paramètres. Le format standard est le suivant : NACA *XYZZ*. Ces paramètres sont des grandeurs physiques ayant des liens directs avec les performances aérodynamiques. Il s'agit de :

X Cambrure maximum en centième de corde

Y Position de la cambrure maximale en dizaine de % de corde

ZZ Épaisseur du profil normal à la ligne de cambrure

Par exemple, un NACA 0012, est un profil symétrique (ligne de cambrure nulle) et a une épaisseur maximale égale à 12% de la corde. Un NACA 2408 a une cambrure maximale de 2% de corde située à 40% du bord d'attaque et une épaisseur de 8%. La figure

1.3 illustre des exemples de profils NACA. Dans la figure, deux profils ayant la même épaisseur sont représentés pour illustrer l'effet de la cambrure.

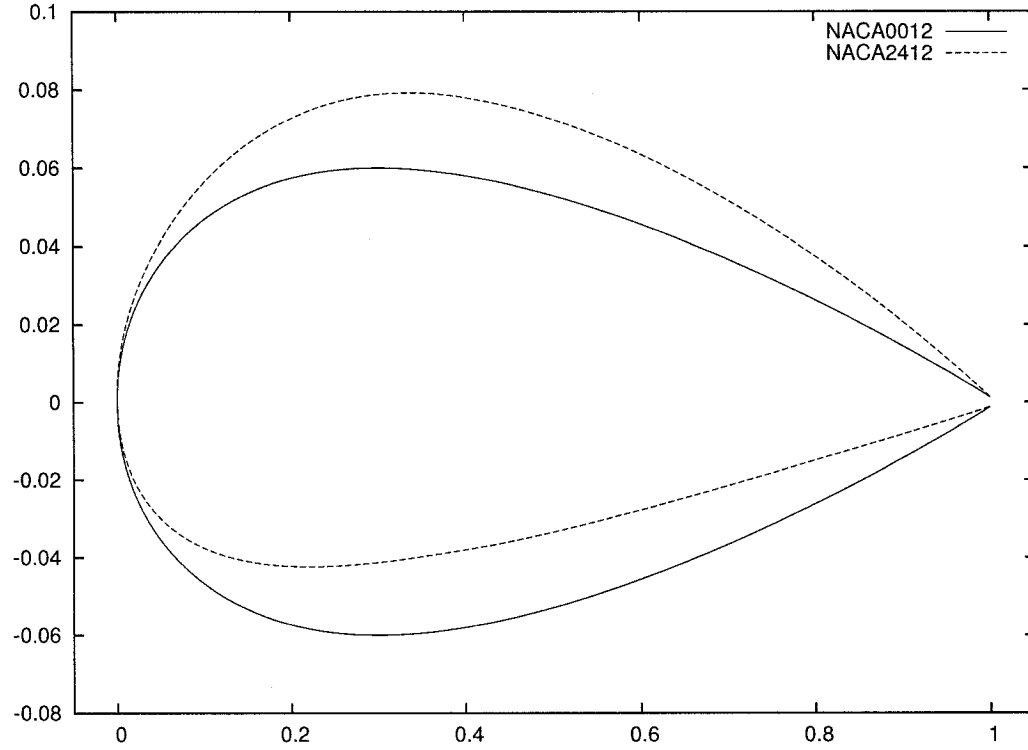


FIG. 1.3: Profil NACA 0012 et 2412

L'intrados et l'extrados du profil sont générés en plusieurs étapes. La philosophie de construction des profils NACA est d'ajouter une épaisseur à une ligne de cambrure moyenne. Il faut d'abord calculer cette dernière et lui ajouter l'épaisseur. Les coordonnées de l'extrados (x_{ex}, y_{ex}) et de l'intrados (x_{in}, y_{in}) sont alors respectivement données par :

$$\begin{aligned} x_{ex}(x) &= x - y_t(x) \sin(\theta(x)) & \text{et} & & y_{ex}(x) &= y_c(x) + y_t(x) \cos(\theta(x)) \\ x_{in}(x) &= x + y_t(x) \sin(\theta(x)) & \text{et} & & y_{in}(x) &= y_c(x) - y_t(x) \cos(\theta(x)) \end{aligned} \quad (1.5)$$

où la ligne de cambrure moyenne y_c est donné par :

$$y_c(x) = \begin{cases} \frac{X}{Y^2}(2Yx - x^2) & \text{si } x \leq Y \\ \frac{X}{1 - Y^2}((1 - 2Y) + 2Yx - x^2) & \text{si } x \geq Y \end{cases} \quad (1.6)$$

et la demi épaisseur y_t s'écrit :

$$y_t(x) = \frac{ZZ}{0.20}(0.2969 \sqrt{x} - 0.126x - 0.3516x^2 + 0.2843x^3 - 0.1015x^4) \quad (1.7)$$

Enfin, θ est la pente de la ligne de cambrure moyenne donnée par :

$$\theta = \tan^{-1} \left(\frac{dy_c(x)}{dx} \right) \quad (1.8)$$

où $x \in [0, 1]$.

La paramétrisation de la famille NACA quatre chiffres a grandement été utilisée. Beaucoup de leurs profils définissent la géométrie d'ailes ou de gouvernes d'avions étant passés à l'histoire. Cependant, avec l'augmentation continue de la vitesse de croisière des avions, ceux-ci font face à de nouveaux comportements de l'écoulement, en entrant dans le régime transsonique. Des profils, issus du génie de Whitcomb (Summerfield 1982), sont conçus pour faire face à de telles situations. Cependant, la famille à quatre chiffres n'est pas adaptée au régime transsonique. En effet, ils ne permettent pas d'obtenir les principales caractéristiques des profils supercritiques, qui sont caractérisés par un grand rayon de courbure au bord d'attaque, un extradors plat et une inflexion vers l'arrière de l'intrados (voir section 2.2).

Autres familles NACA

Bien que la famille NACA *4-digit* soit la plus connue et répandue, il n'en demeure pas moins que c'est une famille désuète pour faire face aux conditions de vol d'aujourd'hui.

C'est pourquoi la NASA a développé d'autres familles de profils, comme les NACA *5-digit* et *6-digit* et les familles de profils supercritiques (Harris 1990).

La famille *5-digit*, développée dans les années 1930, ne s'est jamais tellement répandue, seul le NACA 23015 a vraiment été exploité. Ici aussi, le premier chiffre et les deux derniers représentent respectivement la cambrure et l'épaisseur maximales. Le deuxième représente toujours la position de la cambrure maximum, mais est exprimé en intervalle deux fois plus petit. Le chiffre supplémentaire, le 0 dans l'exemple, est inséré au milieu et indique que la ligne de cambrure est droite à l'arrière du point de cambrure maximum. Un 1 aurait indiqué que la ligne de cambrure est courbée, comme les NACA4-*digit*.

La famille *6-digit* est aussi connue sous le nom de profil à écoulement laminaire (*laminar flow airfoil*). Cette famille a connu un vif succès. La nomenclature de cette série est relative aux caractéristiques aérodynamiques, plus précisément aux coefficients de pressions et de portance. Le premier chiffre n'indique que la famille. Le second indique l'endroit où le coefficient de pression est minimum, en demi pourcentage de corde. Le troisième représente la grandeur de la zone où les hypothèses de faible traînée peuvent être valide, en dixième d'unité de coefficient de portance. Le quatrième est le coefficient de portance proprement dit, en dixième d'unité. Enfin, les deux derniers représentent toujours l'épaisseur maximale en pourcentage de corde. Par exemple, le profil 65₂ – 415, est un profil de la famille à 6 chiffres, dont l'épaisseur est égale à 15% et dont le point de pression minimum est situé à 2.5% de la corde. Le coefficient de portance de design du profil est de 0.4 et la traînée demeurent faible pour une variation du c_l de ± 0.2 .

Enfin, les familles de profils supercritiques, ont été développées suite aux travaux de Whitcomb (Summerfield 1982). Ce dernier, a développé en 1965 le premier profil supercritique. On les désigne sous le nom NACA SC. Ici encore, le coefficient de portance de design et l'épaisseur se retrouvent codés.

Bien qu'un grand savoir ait été accumulé au fil des ans, les familles de profils NACA ont comme principal inconvénient d'être pleinement définies par leur numéro. Ceci implique qu'il ne reste pas de paramètre pour modifier un profil. La formulation mathématique de ces profils ne permet pas non plus de modifications locales, ce qui est un grand inconvénient pour l'optimisation.

1.1.3 Approche discrète

L'approche discrète, comme son nom l'indique, s'appuie sur la discrétisation de la géométrie. Cette méthode consiste à prendre les coordonnées de la série de points frontières représentant la géométrie comme variables de design. Plusieurs auteurs ont utilisé cette méthode pour faire des optimisations : mentionnons entre autre Campbell (1992) et Jameson (1994). Cette approche peut également être employée pour la représentation de surface, comme l'a fait Jameson. Pour l'étude de ce dernier, la paramétrisation était basée sur la position des points dans le maillage. Plus de 4000 variables de design ont été utilisées, ce qui est un très grand nombre pour ce type de problème.

Cette technique est facile à implémenter. Les déformations que la géométrie peut subir sont seulement limitées par le nombre de variable de design et, en contraignant efficacement le mouvement des points, il est possible d'avoir un contrôle très local de la forme.

Cependant, avec un grand nombre de paramètres, il est difficile de maintenir une géométrie lisse. Ceci a comme conséquence que le résultat d'une optimisation peut ne pas être manufacturable. Pour contrôler les oscillations, des contraintes multipoints, des ajustements dynamiques des bornes de chaque variable de design ou encore des algorithmes de lissages ont été suggérés. Due au grand nombre de variables de design, cette méthode n'a été utilisée que conjointement avec les méthodes adjointes, qui permettent le calcul des gradients à un coût machine indépendant du nombre de variables de design.

1.1.4 Paramétrisation PARSEC

Une paramétrisation efficace de profils que nous présentons provient des travaux de Sobieczky (1998). Cette paramétrisation est efficace, car son utilisation est très intuitive et son implémentation facile. De plus, cette paramétrisation permet de représenter une multitude de profils, tant conventionnels que supercritiques, et ce, malgré le faible nombre de paramètres de design impliqué dans sa construction. Notons aussi que c'est avec l'aide de cette paramétrisation que le premier prototype de module de génération de la surface d'aile a été implanté (voir chapitre 4).

Cette paramétrisation utilise onze paramètres pour représenter le profil. Ce dernier est représenté sous une forme adimensionnelle définie de $x = 0$ à $x = 1$. Cette paramétrisation utilise une fonction de type explicite $y = f(x)$. En fait, deux fonctions, une pour l'intrados et l'autre pour l'extrados, sont nécessaires à la définition complète du profil. On obtient donc les élévations en fonction de la position x sur la corde. Un des avantages de cette paramétrisation est qu'elle permet de faire aisément le passage des paramètres géométriques aux coefficients mathématiques. En effet, comme les paramètres de construction sont de nature géométriques, leur influence sur la forme du profil est très facile à saisir.

Voici une liste de ces paramètres suivie d'une brève description. La figure 1.4 les illustre tous.

R_{LE} : Rayon de courbure du bord d'attaque

X_{IN} : Localisation du point minimum sur le profil, $0 \leq X_{IN} \leq 1$

Z_{IN} : Point minimum sur le profil, localisé en X_{IN}

R_{IN} : Rayon de courbure au point minimum

X_{EX} : Localisation du point maximum sur le profil, $0 \leq X_{EX} \leq 1$

Z_{EX} : Point maximum sur le profil, localisé en X_{EX}

R_{EX} : Rayon de courbure au point maximum

θ : Angle moyen du bord de fuite avec l'horizontal

ϕ : Angle de coin du bord de fuite

Z_{TE} : Position du bord de fuite à $x = 1$

ΔZ_{TE} : Épaisseur du bord de fuite ($0 \rightarrow$ profil fermé)

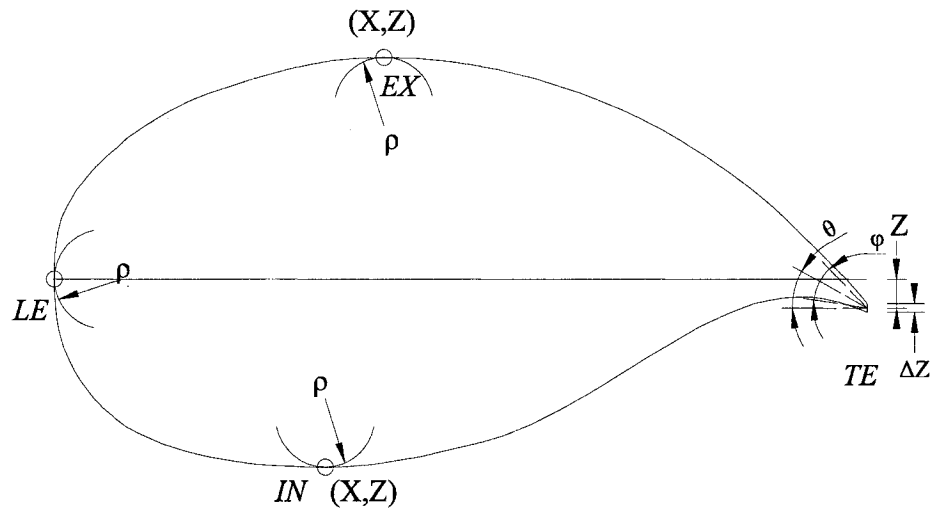


FIG. 1.4: Paramètres utilisés dans la représentation PARSEC

À partir de ce jeu de paramètres, il faut trouver les coefficients a_i et b_i (eq. 1.9) des deux fonctions représentant l'intrados et l'extrados. La fonction a la même forme pour représenter ces deux parties du profil, seuls les coefficients changent. La fonction est une série de puissance, tout comme certains profils NACA (voir section 1.1.2), mais son ordre est plus élevé. Les équations régissant l'intrados et l'extrados sont respectivement :

$$\begin{cases} y_i(x) = \sum_{i=0}^5 a_i x^{(i+1/2)}, \text{ pour } x \in [0, 1] \\ y_e(x) = \sum_{i=0}^5 b_i x^{(i+1/2)}, \text{ pour } x \in [0, 1] \end{cases} \quad (1.9)$$

Il faut donc obtenir les six coefficients a_i qui déterminent l'intrados et les six b_i pour l'extrados à partir des paramètres. On remarque qu'il y a onze paramètres pour définir douze coefficients. L'apparente incohérence provient du fait que le paramètre R_{LE} est utilisé pour l'intrados et l'extrados. On remarque aussi que tous ces paramètres sont appliqués en seulement cinq points distincts : au bord d'attaque (LE), aux points minimum (IN) et maximum (EX) et au bord de fuite sur l'intrados (TE_{in}) et sur l'extrados (TE_{ex}). Les positions et les angles sont imposés à tous ces points et les courbures sur les trois premiers. Enfin, notons que dû au fait que la pente est infinie au bord d'attaque et que la paramétrisation s'appuie sur des fonctions explicites, il peut y avoir des problèmes lors de l'évaluation numérique de la fonctions ou de ses dérivées en ce point.

Il faut assembler deux systèmes d'équations pour faire le passage des paramètres géométriques aux coefficients a_i et b_i . Chacun de ces systèmes détermine les six coefficients relatifs à sa fonction associée. Pour ce faire, il faut d'abord se doter d'une relation entre la courbure κ d'une courbe en un point et ses dérivées, ce que donne la relation 1.10 :

$$\kappa = \frac{\frac{d^2y}{dx^2}}{\left(1 + \left(\frac{dy}{dx}\right)^2\right)^{3/2}} \quad (1.10)$$

Ici, les dérivées des fonctions représentant la forme du profil sont :

$$y'_{in/ex}(x) = \sum_{i=0}^5 c_i(i + 1/2)x^{(i-1/2)} \quad (1.11)$$

$$y''_{in/ex}(x) = \sum_{i=0}^5 c_i(i + 1/2)(i - 1/2)x^{(i-3/2)} \quad (1.12)$$

$$\text{avec } c_i = \begin{cases} a_i & \text{pour l'intrados} \\ b_i & \text{pour l'extrados} \end{cases}$$

Bord d'attaque

Une propriété intéressante de cette paramétrisation est que certaines caractéristiques désirées sur le profil sont obtenues par construction. Comme on décrit des profils adimensionnels ($0 \leq x \leq 1$), il faut que le profil passe par l'origine et qu'il ait une pente infinie à cet endroit. Or, la définition du profil par l'équation 1.9 induit ces deux caractéristiques par construction.

En ce qui concerne la courbure, il faut l'imposer explicitement. En substituant 1.10 dans les équations 1.11 et 1.12 et en faisant un passage à la limite ($\lim x \rightarrow 0$), il est possible de simplifier l'expression pour imposer R_{LE} . En procédant ainsi, on trouve :

$$R_{LE} = \frac{2}{a_0^2} = -\frac{2}{b_0^2} \quad (1.13)$$

On remarque alors que le paramètre R_{LE} fixe le premier coefficient dans les deux systèmes. Le signe inversé des courbures indique que les deux courbes tournent en sens contraire au bord d'attaque.

Points minimum et maximum

On connaît l'élévation et la position sur la corde de celle-ci pour les points minimum et maximum du profil recherché. Il faut donc contraindre les courbes à passer par ces deux points aux bonnes valeurs de x . Ceci donne, pour l'intrados et l'extrados :

$$\begin{cases} y_{in}(X_{IN}) = \sum_{i=0}^5 a_i X_{IN}^{i+1/2} = Z_{IN} \\ y_{ex}(X_{EX}) = \sum_{i=0}^5 b_i X_{EX}^{i+1/2} = Z_{EX} \end{cases} \quad (1.14)$$

De plus, comme ce sont des élévation minimale et maximale, on sait que les dérivées premières à ces endroits sont nulles, d'où :

$$\begin{cases} y'_{in}(X_{IN}) = \sum_{i=0}^5 a_i(i + 1/2)X_{IN}^{i-1/2} = 0 \\ y'_{ex}(X_{EX}) = \sum_{i=0}^5 b_i(i + 1/2)X_{EX}^{i-1/2} = 0 \end{cases} \quad (1.15)$$

Enfin, toujours en ces deux endroits, à l'aide de l'équation 1.10 et en substituant $y'(X_{IN}) = y'(X_{EX}) = 0$, on trouve :

$$\begin{cases} y''_{in}(X_{IN}) = \sum_{i=0}^5 a_i(i + 1/2)(i - 1/2)X_{IN}^{i-3/2} = -R_{IN} \\ y''_{ex}(X_{EX}) = \sum_{i=0}^5 b_i(i + 1/2)(i - 1/2)X_{EX}^{i-3/2} = R_{EX} \end{cases} \quad (1.16)$$

Bord de fuite

La position moyenne du bord de fuite est donnée par Z_{TE} . Habituellement, un profil est défini de l'origine au point $(1, 0)$, la hauteur du bord de fuite étant donc zéro. Cette formulation donne plus de flexibilité en permettant de choisir la hauteur du bord de fuite. Cependant, on suggère de ne pas entrer une valeur différente de zéro, car faire de la sorte inclut artificiellement un angle d'incidence α au profil et modifie légèrement sa géométrie. Une discussion sur ce problème est élaborée au chapitre 4. Cette paramétrisation donne également la possibilité de fixer l'épaisseur du bord de fuite ΔZ_{TE} . Pour obtenir un profil fermé, il suffit d'imposer cette valeur à zéro, sinon l'épaisseur est distribuée en deux parties égales sur l'intrados et l'extrados. Les conditions à imposer sont donc :

$$\begin{cases} y_i(1) = \sum_{i=0}^5 a_i = Z_{TE} + \frac{\Delta Z_{TE}}{2} \\ y_e(1) = \sum_{i=0}^5 b_i = Z_{TE} - \frac{\Delta Z_{TE}}{2} \end{cases} \quad (1.17)$$

Les pentes de l'intrados et l'extrados au bord de fuite peuvent être trouvées aisément avec α et β . On les impose comme suit :

$$\begin{cases} y'_e(1) = \sum_{i=0}^5 (i + 1/2) a_i = \tan(\alpha + \frac{\beta}{2}) \\ y'_i(1) = \sum_{i=0}^5 (i + 1/2) b_i = \tan(\alpha - \frac{\beta}{2}) \end{cases} \quad (1.18)$$

Système d'équation

Avec les équations 1.13 à 1.18, les deux systèmes d'équations peuvent être construits. Il sont donnés par :

$$\frac{1}{4} \begin{bmatrix} 4 & 0 & 0 & 0 & 0 & 0 \\ 4 & 4 & 4 & 4 & 4 & 4 \\ 4X_{IN/EX}^{1/2} & 4X_{IN/EX}^{3/2} & 4X_{IN/EX}^{5/2} & 4X_{IN/EX}^{7/2} & 4X_{IN/EX}^{9/2} & 4X_{EXIN}^{11/2} \\ 2X_{IN/EX}^{-1/2} & 6X_{IN/EX}^{1/2} & 10X_{IN/EX}^{3/2} & 14X_{IN/EX}^{5/2} & 18X_{IN/EX}^{7/2} & 22X_{EXIN}^{9/2} \\ -X_{IN/EX}^{-3/2} & 3X_{IN/EX}^{-1/2} & 15X_{IN/EX}^{1/2} & 35X_{IN/EX}^{3/2} & 77X_{IN/EX}^{5/2} & 143X_{EXIN}^{7/2} \\ 2 & 6 & 10 & 14 & 22 & 30 \end{bmatrix} \begin{bmatrix} a_0/b_0 \\ a_1/b_1 \\ a_2/b_2 \\ a_3/b_3 \\ a_4/b_4 \\ a_5/b_5 \end{bmatrix} = \begin{bmatrix} \sqrt{\pm 2/R_{LE}} \\ Z_{TE} \pm \Delta Z_{TE}/2 \\ Z_{IN/EX} \\ 0 \\ R_{IN/EX} \\ \tan(\alpha \pm \beta/2) \end{bmatrix} \quad (1.19)$$

La résolution de ces derniers va permettre de déterminer les coefficients des polynômes qui définissent le profil. La figure 1.5 montre quelques exemples de profils représentés avec la paramétrisation PARSEC. On y constate l'effet qu'ont sur la courbe les points de contraintes. L'allure de la courbe est très régulière entre ceux-ci. Dans ces exemples, un profil de base est modifié pour montrer l'effet de certains paramètres. Le rayon de courbure au bord d'attaque est augmenté puis diminué en tireté (- - -) sur la figure du haut. Les points maximum et minimum sont bougés sur la figure du bas. Quant aux paramètres gouvernant le bord de fuite, ils sont laissés constant pour souligner leurs effets sur la courbe. En effet, tous les profils se confondent à partir de 95% de corde.

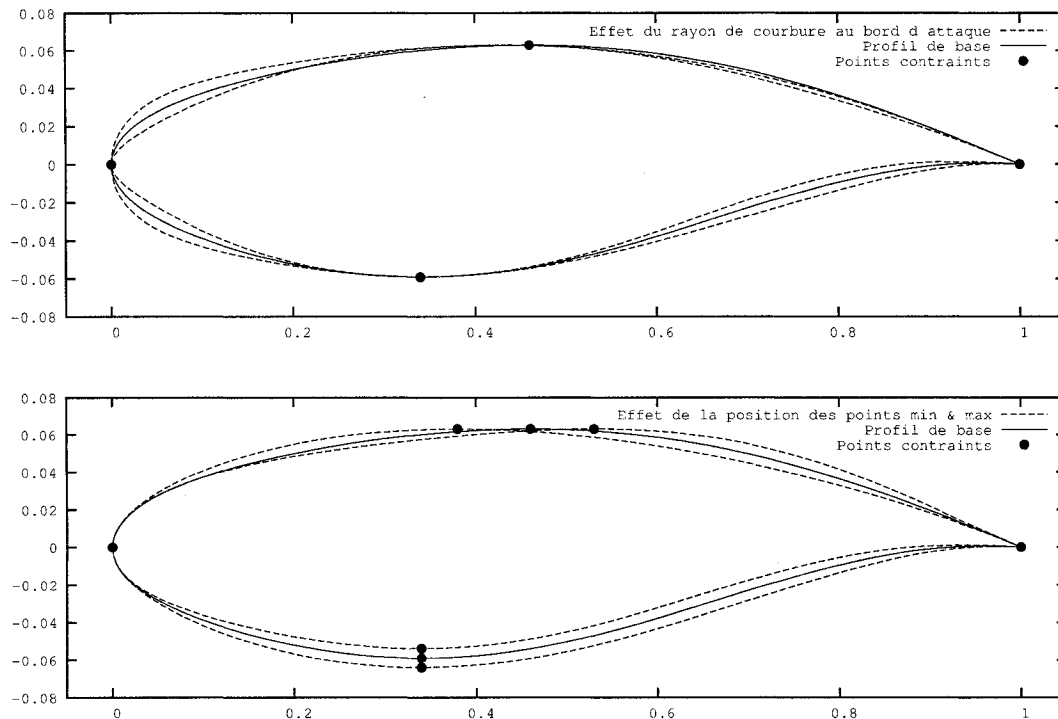


FIG. 1.5: Exemple de profils créés avec la paramétrisation PARSEC. Effet du rayon de courbure au bord d'attaque (haut) et de la position des points minimum et maximum (bas)

Cependant, une limite de cette paramétrisation est le contrôle local. En effet, on constate que changer un seul paramètre affecte toute une moitié de profil, et tout le profil dans le cas du rayon de courbure au bord d'attaque. Cette paramétrisation ne permet pas les changements de forme locaux.

1.1.5 Bases polynomiales pour la représentation de courbe

L'utilisation de fonctions polynomiales est bien adaptée pour la représentation de courbes. Un polynôme peut décrire des courbes complexes sous une forme très compacte, donc avec un faible nombre de variables de design. Les courbes polynomiales donnent une représentation lisse dans presque tout les cas, sans inclure explicitement des contraintes

sur la continuité.

Avant d'aller plus loin, il faut bien faire la distinction entre une fonction (représentation explicite & implicite) et une courbe paramétrique. Ici, contrairement au cas précédents (PARSEC & NACA) où l'on travaillait avec des fonctions explicites, la courbe est paramétrique. Une courbe paramétrique se distingue d'une fonction sur plusieurs points. Premièrement, toutes les composantes sont fonction d'un autre paramètre (d'où le nom courbe paramétrique). Au lieu d'avoir une courbe tel que $y = f(x)$ ou $f(x, y) = cte$, on a $\mathbf{C}(u) = (x(u), y(u))$. La courbe est générée en calculant les valeurs des composantes en fonction du paramètre u . Usuellement, ce dernier est compris entre 0 et 1 et représente en quelque sorte l'avancement sur la courbe. Les avantages des courbes paramétriques sont, entre autres, la possibilité de faire des courbes fermées, avec des points de rebroussement ou avec une pente infinie.

Une courbe polynomiale paramétrique est décrite de manière générale par :

$$\mathbf{R}(u) = \sum_{i=0}^{n-1} \mathbf{c}_i u^i \quad (1.20)$$

où $\dim(c) \times n$ est le nombre de variable de design et u est le paramètre de parcours de la courbe. Les \mathbf{c}_i sont des ensembles de vecteurs de coefficients. Les $\dim(c)$ composantes de ces vecteurs sont utilisées comme variables de design. La dimension est égale à 2 ou 3 selon que l'on se trouve dans le plan ou l'espace.

Cette forme est compacte mais n'est guère pratique. En effet, avec cette représentation, les coefficients ne conservent aucune propriété géométrique. Il est donc difficile de manipuler ces courbes par leurs coefficients (voir fig. 1.6). De plus, le contrôle local n'est pas supporté. Si on modifie la grandeur d'un seul coefficient, c'est toute la courbe qui est affectée. Aussi, la formulation avec une série de puissance peut mener à des erreurs d'arrondis, surtout si les coefficients ne sont pas du même ordre de grandeur. Enfin, lorsque

l'on emploie un degré élevé, certains comportements bizarres peuvent apparaître, comme de fortes oscillations. Il n'en reste pas moins que cette formulation est compacte et peut s'avérer utile pour des courbes relativement simples.

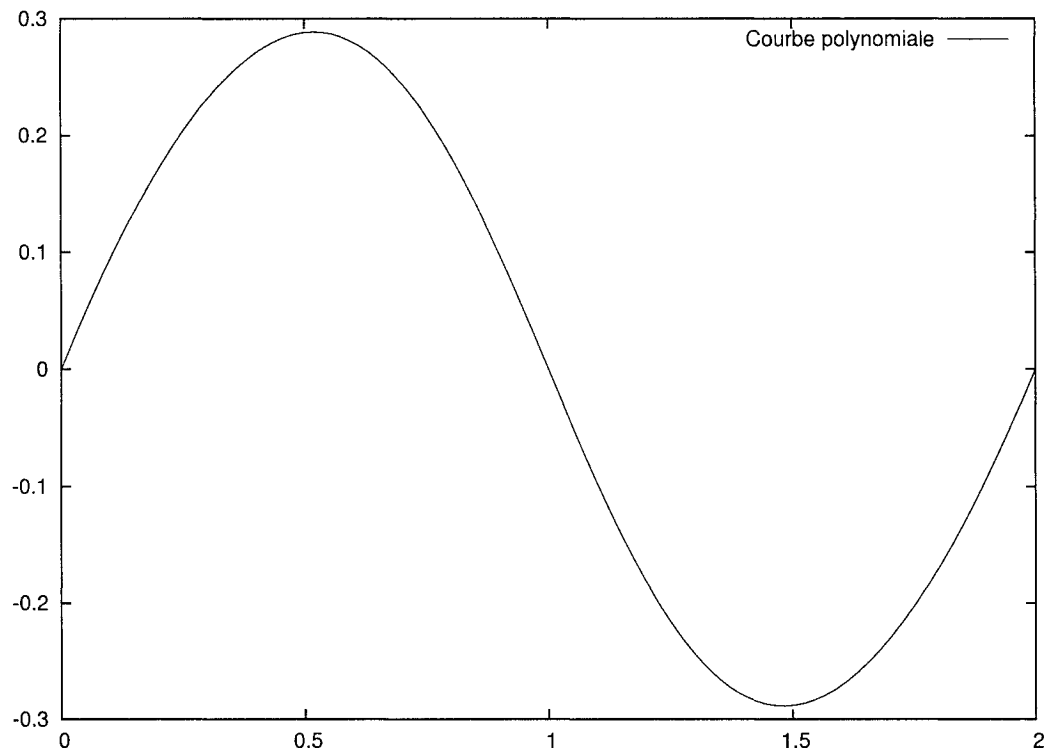


FIG. 1.6: Courbe polynomiale de degré 3, avec $\mathbf{c}_0 = [0 \ 0]$, $\mathbf{c}_1 = [3 \ 3]$, $\mathbf{c}_2 = [-3 \ -9]$ et $\mathbf{c}_3 = [2 \ 6]$ exprimée sur $u \in [0, 1]$.

1.1.6 Courbes de Bézier

La représentation à l'aide de courbes de Bézier est un équivalent des bases polynomiales. C'est une autre formulation mathématique pour représenter des courbes, plus facile à manipuler que la précédente. Notons l'ajout de points de contrôle dans la formulation. Ceux-ci, par leurs positions, donnent au concepteur l'allure générale de la courbe. Une

courbe de Bézier est décrite par :

$$\mathbf{P}(u) = \sum_{i=0}^n B_{i,n}(u) \mathbf{P}_i \quad (1.21)$$

où les \mathbf{P}_i sont les $n + 1$ points de contrôle et les $B_{i,n}$ sont les polynômes de Bernstein de degré $p = n$ donnés par :

$$B_{i,n}(u) = \frac{n!}{i!(n-i)!} u^i (1-u)^{n-i} \quad (1.22)$$

Cette courbe est définie sur l'intervalle $u \in [0, 1]$. Cette formulation, bien qu'elle soit l'équivalent mathématique de la représentation sous forme de série de puissance, est mieux adaptée à la manipulation de courbes. Les positions des points de contrôle constituent les variables de design et forment le polygone de contrôle. On voit, sur la figure 1.7, qu'ils approximent l'allure de la courbe et qu'ils donnent la pente aux extrémités. Cette caractéristique donne une meilleure idée de l'influence des variables de design sur la courbe. Aussi, l'évaluation numérique d'une courbe de Bézier est très efficace avec l'emploi de l'algorithme de deCasteljau (Piegl et Tiller 1995). L'enveloppe convexe sous tendue par le polygone de contrôle contient toujours la courbe et cette dernière interpole toujours le premier et dernier point de contrôle, respectivement à $u = 0$ et $u = 1$.

Cependant, comme il n'y a qu'un seul intervalle paramétrique ($0 \leq u \leq 1$) et que les fonctions de bases ($B_{i,n}(u)$) ne sont jamais nulles à l'intérieur de ce dernier, les points de contrôle influencent toute la courbe. Par conséquent, comme dans les cas précédents, le contrôle local n'est pas supporté.

La représentation avec une courbe de Bézier est adaptée pour des courbes relativement simples. En augmentant la complexité de la courbe à représenter, le nombre de points de contrôle augmente et forcément le degré de la courbe. Or, plus le degré est élevé, plus grands sont les risques d'oscillations. On remarque donc qu'il n'est pas utile d'em-

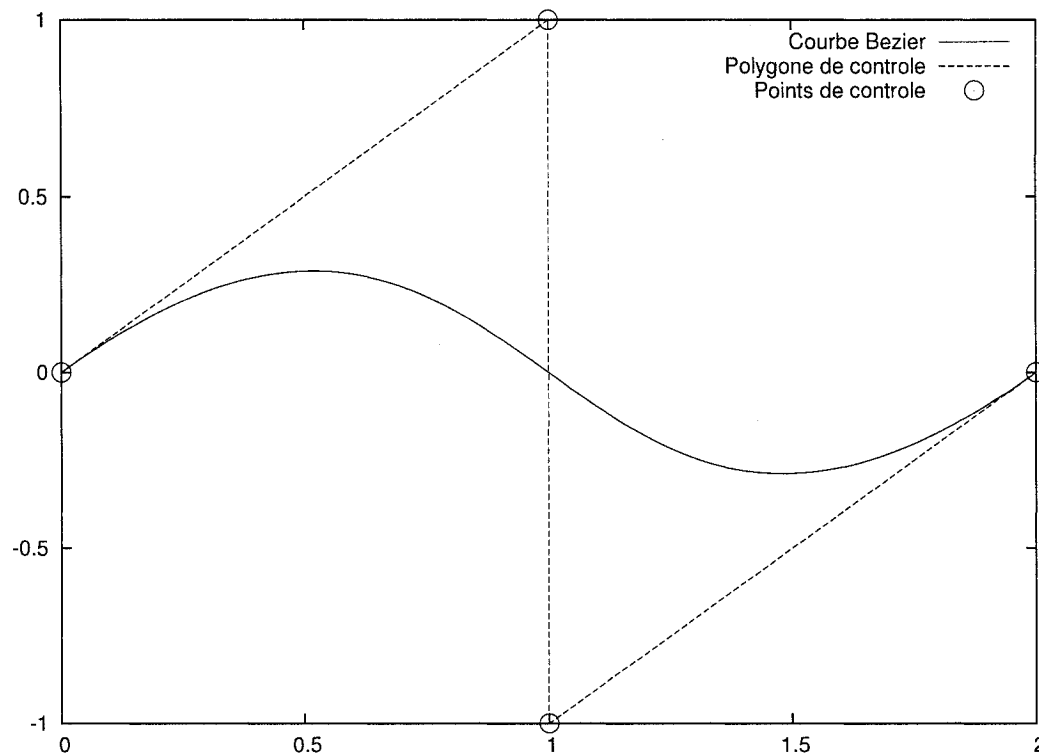


FIG. 1.7: Courbe de Bézier, identique à la précédente (fig. 1.6, échelle différente), avec 4 points de contrôle ; $P_0 = (0, 0)$, $P_1 = (1, 1)$, $P_2 = (1, -1)$ et $P_3 = (2, 0)$

ployer une Bézier d'un degré très élevé. Si la courbe à représenter est complexe, d'autres approches doivent être envisagées.

Il est possible entre autre d'utiliser plusieurs courbes de Bézier raccordées. Cependant, en agissant de la sorte, si on veut garder une certaine continuité à la jonction des morceaux de courbes, on doit ajouter explicitement des conditions de raccord à ces endroits. Pour imposer ces contraintes, il faut résoudre un système linéaire pour le positionnement des points de contrôle. Ceux-ci deviennent donc fixés par ces conditions et ne peuvent plus bouger sans compromettre la continuité de la courbe résultante.

1.1.7 Courbes B-Splines & NURBS

Nous allons maintenant présenter le prolongement naturel des courbes de Bézier, les B-Splines. Une B-Spline est l'équivalent d'un collage de courbes de Bézier avec un support mathématique plus attrayant. Elles ont la forme suivante :

$$\mathbf{C}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i \quad (1.23)$$

où les \mathbf{P}_i sont les $n + 1$ points de contrôle et $N_{i,p}(u)$ est la $i^{\text{ème}}$ fonction de base B-Splinaire de degré p . Les fonctions de bases sont entièrement définies à l'aide du degré de la courbe et du vecteur nodal \mathbf{U} . Ce dernier contient m nœuds, les u_i . Le nombre de point de contrôle n , le degré p et le nombre de nœuds m sont reliés par la relation $m = n + p + 1$. Les nœuds du vecteur nodal sont toujours monotones croissants, *i.e.* $u_i \leq u_{i+1}$

Les bornes inférieures et supérieures du vecteur nodal sont généralement égales à zéro et un. Les nœuds intérieurs ($0 < u < 1$) divisent l'intervalle paramétrique en plusieurs sous intervalles et par conséquent la courbe en plusieurs segments. Chacun de ces segments est l'équivalent d'une courbe de Bézier, un peu comme s'il y avait $n - p + 1$ courbes de Bézier de degré p raccordées ensemble. Dans cette représentation, contrairement à celle utilisant des courbes de Bézier, le degré p de la courbe et le nombre de points de contrôle n sont indépendants. Cependant, la relation ($p \leq n$) doit toujours être respectée et dans le cas où $p = n$, la B-Spline devient une courbe de Bézier.

En plus de toutes les caractéristiques intéressantes de la représentation Bézier, les B-Splines de faible ordre peuvent aisément représenter des courbes très complexes. Cette famille de courbe a toutefois ses limites : elle ne permet pas de représenter les coniques. Cependant, la forme rationnelle des B-Spline, les NURBS, permet de contourner ce problème. Il s'agit d'associer un poids à chacun des points de contrôle. Une courbe

NURBS est définie par :

$$\mathbf{C}(u) = \frac{\sum_{i=0}^n N_{i,p}(u) \omega_i \mathbf{P}_i}{\sum_{i=0}^n N_{i,p}(u) \omega_i} \quad (1.24)$$

Ou, en écriture simplifiée :

$$\begin{aligned} \mathbf{C}(u) &= \sum_{i=0}^n R_{i,p}(u) \mathbf{P}_i \\ \text{avec} \quad R_{i,p}(u) &= \frac{N_{i,p}(u) \omega_i}{\sum_{j=0}^n N_{j,p}(u) \omega_j} \end{aligned} \quad (1.25)$$

où les \mathbf{P}_i sont les $n + 1$ points de contrôle ; $N_{i,p}(u)$ est la $i^{\text{ème}}$ fonction de base B-Splinaire de degré p ; $R_{i,p}(u)$ est la $i^{\text{ème}}$ fonction de base rationnelle de degré p et les ω_i sont les poids associés à chaque point de contrôle. On peut employer une formulation plus légère pour les NURBS grâce à l'emploi des coordonnées homogènes. Dans ce cas, la courbe s'exprime sous la forme suivante :

$$\mathbf{C}^\omega(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i^\omega \quad (1.26)$$

où $\mathbf{P}^\omega = (x\omega, y\omega, z\omega, \omega) = (X, Y, Z, W)$. L'avantage de cette dernière formulation est que le poids d'un point de contrôle est présent comme une composante et que la courbe NURBS peut s'écrire sous forme polynomiale (non rationnelle).

Cette formulation est très efficace et peut donc réduire le nombre de points de contrôle pour exprimer une même courbe. Lépine *et al.* (1999) ont démontré qu'il est possible de représenter des profils avec des tolérances de fabrication avec seulement une douzaine de points de contrôle. Cependant, bien que l'on réduise le nombre de points de contrôle, le nombre de variables de design reste sensiblement le même, car les points de contrôle

NURBS sont d'une dimension supérieure à celle des B-Splines. Il y a un compromis à faire entre le nombre de point de contrôle et l'emprise sur la courbe qu'ont ceux-ci.

Chez BA, l'expérience a montré que, pour un même nombre de variables de design, l'avantage gagné par l'utilisation de plus de points de contrôle est supérieur à celui gagné par l'utilisation de points de contrôle avec des poids. En conséquence, l'utilisation des B-Splines dépasse celle des NURBS. De plus, les algorithmes de manipulation sur les premières sont plus faciles à coder que sur les secondes. L'annexe I présente quelques algorithmes avancés sur les B-Splines.

1.1.8 Fonctions de formes et fonctions de bases orthogonales

Une approche a été proposée par Hicks et Henne (1979) pour paramétrer un profil dans une formulation compacte. Il s'agit de l'addition de fonctions de formes analytiques. La contribution de chaque fonction est pondérée par un facteur. Ce sont ces facteurs qui sont les variables de design. Cette approche, toute simple, dépend de la qualité des fonctions de formes pour obtenir de bons designs. La continuité du profil résultant est limitée aussi par celle des fonctions de formes. Le choix de celles-ci définit aussi l'espace de design. Il est donc possible de ne pas pouvoir représenter certains profils existants avec exactitude.

Cette méthode a été étendue à des cas plus généraux par Chang *et al.* (1995). Dans leurs travaux, ces fonctions de bases sont obtenues à partir des fonctions génératrices des familles NACA à 4 chiffres et SC. Ils ont montré qu'il est possible de représenter exactement un NACA0012 et de bien approximer certains profils supercritiques. Une vingtaine de coefficients sont nécessaires dans ce cas.

Cependant, comme l'on fait remarquer Robinson et Keane (2001), un problème avec cette méthode est que les fonctions de bases utilisées ne sont pas orthogonales. Ce phénomène brise l'unicité de la transformation entre les paramètres et la géométrie. Ils pro-

posent donc une façon de faire pour construire une base orthonormale de fonctions génératrices à partir de la discrétisation de plusieurs profils. Les paramètres sont des facteurs pondérant l'influence de ces fonctions. Dans leur article, ils utilisent des profils de la famille NACA SC comme générateurs de leurs fonctions orthonormales.

Cependant, aucune de ces méthodes ne supporte le contrôle local. En fait, pour y arriver, il faut ajouter des fonctions non nulles uniquement sur un petit intervalle, ce qui correspond alors à l'approche des fonctions de formes.

Méthodes de paramétrisation 3D

Jusqu'à présent, les méthodes décrites ne sont valables que pour la représentation de courbes, généralement en 2D, mais aussi en 3D (ex : B-Spline). Les méthodes suivantes sont plus générales. Elles sont principalement conçues et utilisées pour la représentation de surfaces tridimensionnelles, mais peuvent aussi être utilisées dans le cas de courbes.

1.1.9 Équations aux dérivées partielles

Une méthode compacte et efficace pour paramétrer les surfaces extérieures d'un avion a été proposée par Bloor et Wilson (1995). Il s'agit de considérer la génération de surface comme étant la solution d'un problème elliptique aux dérivées partielles soumis à des conditions frontières. Il a été démontré qu'il est possible de représenter la surface d'un avion générique à l'aide d'un petit nombre de variables de design (une vingtaine). Cependant, certaines variables n'ont pas de signification physique, rendant leur ajustement plutôt difficile, même pour un ingénieur expérimenté.

Cette technique a été étendue par Smith *et al.* (1995) à des configurations complètes d'avions comprenant aussi la génération de maillage surfacique et volumique et le cal-

cul de sensibilité pour les applications CFD. La configuration comprend le fuselage, les ailes, les stabilisateurs verticaux et horizontaux ainsi que les plans canards. Des équations aux dérivées partielles (EDP) doivent être solutionnées pour chaque surface créée.

Cependant, l'utilisation d'EDP pour la paramétrisation d'un modèle est coûteux en temps et en calcul. La génération d'un maillage est très complexe à implémenter pour obtenir de bons résultats. Notons que le contrôle local n'est pas supporté. En effet, en changeant les conditions frontières, c'est toute la solution, et donc la surface, qui change. L'utilisation d'EDP semble donc mieux adaptée pour comparer différentes configurations en phase préliminaire que pour faire de l'optimisation fine.

1.1.10 Outils de CAO

Aujourd'hui, les systèmes CAO ont des capacités étendues. Avec la modélisation solide à l'aide de caractéristiques (*features*), disponible dans la majorité des systèmes CAO modernes, on parvient à capturer l'intention derrière le design. Il est désormais possible de définir certaines caractéristiques avec un paramètre et contrôler la géométrie avec celui-ci. Il est difficile de rendre automatique les échanges d'informations entre un tel logiciel et un optimiseur. Notons aussi la difficulté de paramétrer un modèle CAO existant pour améliorer ses caractéristiques.

Récemment, Nemec *et al.* (2004) ont réussi, à l'aide de l'interface CAPRI (Haines 2000) à interagir directement avec un modèle CAO. En plus de fournir un outil efficace pour la discrétisation de surface, CAPRI peut modifier la valeur des paramètres du modèle CAO. De cette façon, le modèle demeure dans le logiciel CAO mais les manipulations sont effectuées à l'externe. De plus, comme le code de cette application est ouvert, l'utilisation d'outils de différentiation automatique est envisageable afin d'améliorer la recherche d'optimum.

1.1.11 Approche orienté sur les déformations libres

Une approche qui semble prometteuse est l'utilisation de la technologie des déformations libres (*FFD - Free Form Deformation*). Elle est issue du savoir accumulé en animation 3D. Comme elle se base sur la déformation, un objet parent est nécessaire. L'idée de base est d'imaginer un objet pris dans un prisme virtuel de plastique élastique. Lorsque l'on déforme ce dernier, c'est tout son intérieur qui se déforme aussi, et donc, l'objet qu'il contient.

Samareh (1999) a étendu cette idée pour l'appliquer à la MDO. En partant du constat qu'il est long de concevoir un bon modèle géométrique complet (surfaces externes, structure interne, etc.) et des maillages résultants, son idée consiste à paramétrer les changements de géométrie, plutôt que la géométrie elle-même. Les déformations sont obtenues à l'aide des déformations libres. Mais comme l'auteur le fait remarquer, les paramètres déterminant ces dernières ne sont pas des grandeurs physiques, rendant leurs manipulations difficiles. L'approche a donc été adaptée et modifiée par Samareh (2000) pour que les paramètres soient davantage des grandeurs physiques, donc plus facilement manipulables.

Cette approche semble attirante sauf qu'elle doit s'appliquer sur un modèle existant. Son développement a été effectué dans une optique de MDO plus que de représentation géométrique seulement. Sa mise en place nécessite un grand déploiement. Une connaissance approfondie sur les FFD doit aussi être acquise. De plus, les déformations devront respecter des contraintes et ceci amoindrit grandement les avantages de cette méthode.

1.2 Critères de Sélection

Il est très important de se donner des balises pour prendre une décision éclairée quant à la méthode de paramétrisation choisie. Ces balises devront se baser sur des critères importants selon les objectifs de notre application, comme le font remarquer Samareh (1999) et Oyama (2000). Voici une liste des critères d'évaluation qui serviront à évaluer les différentes méthodes de paramétrisation pour notre application.

Paramètres d'ingénierie : Une caractéristique importante est la signification physique des variables de design. Un avantage important est que l'ingénieur aérodynamicien d'expérience puisse se faire une bonne idée des caractéristiques des profils et de l'aile directement à l'aide de ces paramètres. Pour mener de bonnes optimisations à l'aide des méthodes à gradient, il est préférable de détenir une solution près de la solution optimale et il est vraiment plus facile de trouver cette solution initiale si les paramètres contrôlent quelque chose de tangible, *i.e.* que l'ingénieur peut facilement conceptualiser pour l'aider au design.

Automatisé : Le module créé doit être autonome, c'est-à-dire qu'une fois lancé, il devrait savoir comment traiter l'information sans aucune intervention humaine. C'est une condition essentielle puisque le module doit être exécuté à répétition dans le cadre d'une optimisation.

Étendue : La technique de paramétrisation retenue devra pouvoir offrir un grand espace de design. Non pas par un grand nombre de variables de design, mais plutôt par l'ensemble des possibilités des courbes pouvant être générées. Celles-ci doivent, dans notre cas, être capable de reproduire avec précision des profils transsoniques classiques.

Compact : Le nombre de variables de design devrait être conservé le plus bas possible pour bien représenter un profil quelconque. En effet, un grand nombre de variable de design nuit à l'obtention de bonnes solutions par des optimiseurs à gradient,

car la recherche de ligne, étape cruciale de ces méthodes, devient très sensible. On veut aussi contrôler la précision en jouant sur le nombre de variables en fonction de l'application.

Contrôle Local : La paramétrisation doit supporter des changements de géométrie localisés. L'avantage du contrôle local est de pouvoir garder le maximum de similarité entre un profil existant satisfaisant et un nouveau où on veut de légers changements locaux pour améliorer une caractéristique. La dimension du vecteur de design est ainsi diminuée lors d'une optimisation, car peu de paramètres sont autorisés à changer. Ceci résulte en une diminution du temps de l'optimisation.

Consistent : Le module doit pouvoir fournir une définition topologique et géométrique complète en fonction de la discipline cliente. Ce critère est nécessaire pour une optimisation multidisciplinaire complète dans un environnement 3D. Comme le but de cette tâche est davantage de fournir un outil de génération de surface d'aile plutôt qu'un outil complet de MDO, nous assurerons que la formulation inclut l'emplacement des longerons, pour pouvoir générer une structure interne réalisable (autre module), et des surfaces hypersustentatrices, dans le but d'effectuer d'autres analyses (aéroélasticité, marge de sécurité, contrôle, etc.). La consistance ici se résume en l'obtention d'une surface d'aile sur laquelle certains éléments sont identifiés, lesquels pouvant être variés afin d'atteindre une valeur prescrite.

Maillage : Le maillage est un élément clé pour mener à bien de bonnes analyses aérodynamiques. Afin de pouvoir communiquer la géométrie vers d'autres systèmes (maquette digitale, mailleur, etc.), il est nécessaire de pouvoir fournir une discrétisation se prêtant au maillage ou à la reconstruction de la surface.

Lien direct avec la CAO : Plus performant que l'échange d'informations géométriques par une discrétisation, le support mathématique peut être directement échangé au modèle CAO. Nous devons donc utiliser un format standard d'échange de données, comme les formats STEP ou IGES. Cependant Samareh (1999) fait remar-

quer que le dernier format STEP (AP 203) ne serait pas encore assez précis pour des analyses aérodynamiques fiables.

Mise en place aisée : Enfin, la mise en place d'un tel système doit être évaluée du point de vue de la complexité. Il s'agit d'évaluer la facilité à implémenter ce système chez Bombardier Aéronautique. Une formulation utilisant des outils de représentation géométrique déjà développés verra ses difficultés d'implémentation réduites.

On trouve dans le tableau 1.2 la synthèse des besoins en regard des critères d'évaluation. Sur la ligne du haut, les neuf différentes techniques et approches pour la paramétrisation de forme décrites à la section 1.1 sont listés. Viennent sur la colonne de gauche, l'ensemble des critères servant à évaluer ces techniques. Enfin, dans le tableau, un signe + indique qu'un critère est respecté par la méthode concernée, un signe – s'il ne l'est pas et un signe \pm si des compromis doivent être faits pour employer cette méthode.

En attribuant une cote de 1 si le critère est satisfait, de 0 s'il l'est plus ou moins et de –1 s'il ne l'est pas, on constate que les méthodes PARSEC et B-Splinaire semblent les mieux adaptées à notre problème. Tout près, viennent les représentations NACA, à l'aide de fonctions orthogonales, d'outils CAO et des déformations libres. Cependant, ces quatre approches présentent toutes de grands problèmes pour les objectifs de ce travail. La représentation NACA ne sort pas du spectre du connu. Les fonctions de bases orthogonales sont dérivées à partir de profils existants. L'espace de design est donc limité par le choix des profils générateurs. De plus, ces deux techniques ne supportent pas le contrôle local. Quant à l'approche CAO et par déformations libres, les contraintes de temps dus à la mise en place sont problématiques.

Par contre, nous gardons la représentation PARSEC, car sa mise en place est très facile et est prescrite par un faible nombre de paramètres d'ingénierie. Un générateur de surface d'aile avec peu de variables peut rapidement être développé à l'aide de cette représentation. Une évaluation du potentiel de la paramétrisation PARSEC a donc été faite dans le

TAB. 1.1: Tableau synthèse des différentes approches

Critères	Paramétrisation	PARSEC	B-Spline	NACA	CAO	Discrete	Déformations libres	Fonctions orthogonales	Transformations conformes	EDP
Paramètres d'ingénierie		+	±	±	±	-	-	-	-	±
Automatisé		+	+	+	±	±	±	±	+	±
Étendue		±	+	±	+	+	+	±	±	±
Compact		+	±	+	-	-	±	+	±	+
Géométrie lisse		+	+	+	±	-	+	+	±	+
Contrôle local		±	±	-	±	+	±	-	-	-
Consistent		-	±	-	+	+	+	-	-	-
Maillage		+	±	+	±	+	±	±	±	-
Lien direct avec la CAO		-	±	-	+	-	-	-	-	-
Mise en place aisé		+	±	+	-	+	-	±	+	-
Total		4	3	2	1	1	0	-2	-2	-3

cadre de ce travail.

Nous sélectionnons aussi les courbes B-Splines. Ces courbes ont une formulation compacte et élégante permettant, avec un faible ordre, d'approximer des formes complexes. Elles sont continues par construction et se prêtent bien au fait de devoir composer avec un nombre variable de paramètres.

CHAPITRE 2

DESCRIPTION DES BESOINS

Nous allons voir ici les besoins auxquels doit répondre le module de construction d'aile d'avion. D'abord, nous énonçons les objectifs dans le but de bien situer le projet. Les besoins et contraintes pour l'aile entière et ses composantes sont ensuite vus en détail. Pour ce faire, nous discutons d'abord de profils 2D, de leurs caractéristiques aérodynamiques et des paramètres les définissant. Ensuite, nous caractérisons la forme en plan et l'aile tridimensionnelle. Nous parlerons enfin sommairement des requis du programme informatique.

2.1 Généralités

Nous voulons concevoir un logiciel de modélisation géométrique d'aile simplifiée pour les avions civils commerciaux. Le module doit pouvoir générer automatiquement une aile en fonction des différents paramètres d'entrées. Ces paramètres doivent être le plus possible des grandeurs physiques donnant un aperçu de l'influence de la variable de design pour l'ingénieur expérimenté. Nous les appellerons donc *paramètres d'ingénierie* étant donné qu'ils représentent quelque chose de tangible et non seulement une valeur numérique abstraite.

Aussi, le nombre de ces paramètres doit être ajustable selon le degré de précision voulu. Il sera alors possible d'effectuer une série d'optimisation en incorporant de nouveaux paramètres permettant ainsi de trouver un bon design, dans un premier temps, pour le raffiner par la suite. En commençant avec un faible nombre de variables de design ayant

une grande influence sur la surface, le module devrait permettre d'explorer un spectre plus large de designs différents.

Mentionnons aussi que le nombre de paramètres a une influence sur la régularité de la surface. En effet, plus le nombre de variables de design est faible, plus la surface a tendance à être lisse naturellement. Comme l'objectif est d'obtenir une surface lisse, *i.e.* qui ne comporte pas d'oscillations, notre expérience nous amène à penser que toute l'information relative à l'enveloppe de l'aile soit contenue dans une centaine de variables de design.

Cette aile doit répondre également à certains critères de performance (finesse), de solidité (épaisseur) et de manufacturabilité (lisse). En plus de ces critères globaux, l'aile doit aussi satisfaire d'autres caractéristiques qui s'appliquent aux familles de courbes issues des projections de l'aile dans le sens de l'envergure et de la hauteur. Nous obtenons alors une série de sections 2D (profils) et la forme en plan de l'aile. Nous verrons en détail aux sections 2.2 et 2.3 les caractéristiques géométriques que ces courbes doivent respecter.

Il faut aussi, comme le font remarquer Robinson et Keane (2001), avoir une formulation où la transformation bijective entre l'espace paramètre et la géométrie soit unique. Ainsi, un jeu de paramètre ne peut résulter qu'en une seule aile, et *vice versa*.

Finalement, comme les ailes construites seront par la suite analysées sur une base aérodynamique, nous devons garder en tête l'idée de pouvoir facilement générer un maillage autour de celles-ci. Dans cette optique, il sera nécessaire de fournir une bonne discrétisation surfacique de l'aile sur laquelle pourra s'accrocher un maillage volumique.

2.2 Profils 2D

Avant de se lancer dans la description des paramètres des profils adimensionnels, nous allons décrire les profils conventionnels versus ceux dits supercritiques. Ces derniers ont vu leur utilisation se répandre dans les années 70 suite aux avancées effectuées par l'ingénieur Richard T. Whitcomb (Summerfield 1982). Confronté à la constante augmentation de la vitesse de croisière des avions, les profils conventionnels se sont avérés à fournir de médiocres performances en régime transsonique. En effet, l'augmentation de la vitesse de croisière amène des effets indésirables de l'écoulement sur un profil conventionnel, dont le pire est une forte onde de choc résultant en une augmentation considérable de la traînée. Les profils supercritiques pallient à la situation en présentant des caractéristiques leurs permettant d'être efficaces à haute vitesse.

2.2.1 Rappel des caractéristiques des profils

Comme nous le savons, la portance est générée par la diminution de pression sur le coté supérieur du profil (extrados) et l'augmentation sur l'autre coté (intrados). Ce changement de pression est tributaire de la variation de vitesse que subit le fluide localement lorsqu'il interagit avec le profil. On constate qu'une augmentation de la vitesse entraîne une diminution de la pression.

Généralement, on parle davantage de coefficients de pression, c_p , en un endroit sur le profil que de pression absolue, p . Ceux-ci sont exprimés par :

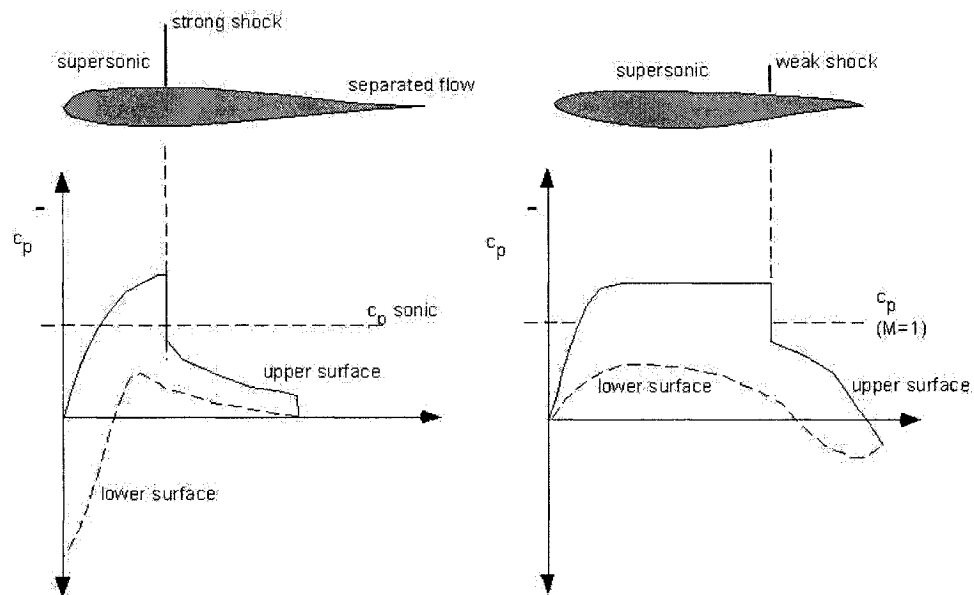
$$c_p = \frac{p - p_\infty}{\frac{1}{2}\rho U_\infty^2} \quad (2.1)$$

où p est la pression locale et p_∞ et U_∞ sont la pression et la vitesse à l'infini. On définit le coefficient de pression critique, $c_{p_{cr}}$, comme le c_p où la vitesse locale de l'écoulement

correspond à $M \geq 1$, où M est le nombre de Mach. L'allure de la distribution des c_p sur un profil nous renseigne sur les caractéristiques aérodynamiques de celui-ci. On peut, en les intégrant, en déduire la portance, la traînée de pression et le moment de force.

Dans le cas du régime transsonique, lorsque la vitesse de l'écoulement à l'infini est autour de $M \sim 0.7-0.8$, une "bulle" supersonique (région de l'espace où la vitesse locale est supérieure à celle du son) se développe sur l'extrados du profil dû à l'accélération du fluide. La zone supersonique, si elle est grande, induit généralement un choc lié à la recompression lors du ralentissement, ce qui génère une traînée d'onde. De plus, si le choc est suffisamment fort, on assiste au décrochage de la couche limite, ce qui génère davantage de traînée de forme. La situation est illustrée à la figure 2.1.

Afin de minimiser ces effets indésirables, les profils supercritiques affichent certaines particularités, non pas sans détériorer les performances à basse vitesse. La forme du profil supercritique est caractérisée par (Summerfield 1982) :



From Bertin & Smith, 2nd Edition, p. 366

FIG. 2.1: Différence aérodynamique entre profils conventionnels et supercritiques [crédit Bertin et Smith (1998)]

Un grand rayon de courbure au bord d'attaque : Afin de diminuer l'amplitude du choc, voir de le faire disparaître, les coefficients de pression sur les profils supercritiques sont moins négatifs sur l'extrados que sur les profils conventionnels. Le grand rayon de courbure permet cet effet en diminuant la portance dans la partie avant du profil.

Une zone plate sur l'extrados : Cette zone presque droite permet de retarder l'apparition du choc sur le profil. Bien que les c_p vers l'avant soient augmentés, ceux vers l'arrière diminuent. En conséquence, on note l'apparition d'un plateau dans les valeurs de c_p dans cette zone.

Une cavité sur la partie arrière de l'intrados : Cette caractéristique a pour objectif de récupérer le plus possible la portance qui a été perdue dans la partie avant. La cavité permet d'augmenter les c_p vers l'arrière par la présence d'un gradient de pression adverse. L'aire de la courbe des c_p s'en trouve augmentée, et donc, la portance. Ceci induit par contre des moments plus grands.

2.2.2 Les paramètres des profils 2D

Considérons un profil aérodynamique 2D dans le plan xz et normalisé dans l'intervalle $0 \leq x \leq 1$. Le bord d'attaque correspond à l'origine et le bord de fuite est positionné à $(1, 0)$. Comme nous voulons représenter des profils supercritiques, notre définition doit supporter les caractéristiques inhérentes à ce type de section.

D'abord, les profils seront définis par des courbes construites à l'aide d'un jeu de paramètres. Ces paramètres doivent être des grandeurs ayant une signification physique, géométrique ou aérodynamique. Voici une liste de ces paramètres candidats, tel que présentés à la figure 2.2 :

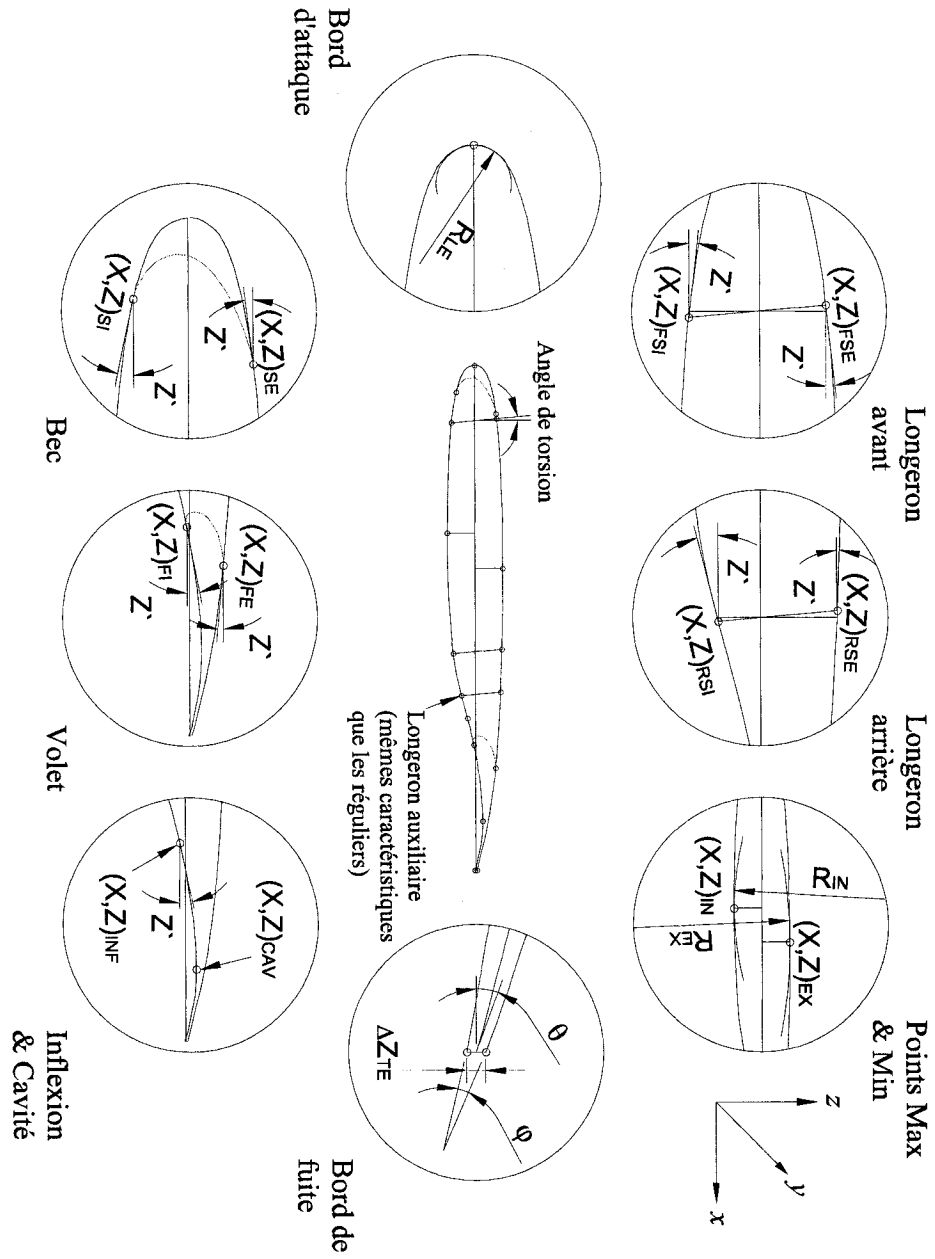


FIG. 2.2: Identification des paramètres géométriques pour définir un profil aérodynamique 2D

1. Bord de fuite

- (a) Angle moyen par rapport à l'axe de la corde (θ).
- (b) Angle inclus (φ).
- (c) Épaisseur (ΔZ_{TE}).

2. Bord d'attaque

- (a) Rayon de courbure R_{LE}
- (b) Un autre paramètre pour contrôler "l'épaisseur" du bord d'attaque (non représenté sur la figure). Dans le passé, la communauté scientifique a utilisé un paramètre ΔZ_{LE} qui est la position de l'extrados à $x/c = 6\%$ moins la position de l'extrados à $x/c = 1.5\%$.

3. Points maximum et minimum

- (a) Hauteur maximale et minimale de l'extrados et de l'intrados (Z_{EX}, Z_{IN})
- (b) Emplacement sur la corde de ces hauteurs (X_{EX}, X_{IN})
- (c) Rayon de courbure à ces deux points (R_{EX}, R_{IN})

4. Points aux longerons

- (a) Emplacement X des longerons, (X_{FS}, X_{RS})
- (b) Hauteur (Z) des 4 points définis par le dessus et le dessous des longerons avant et arrière ($Z_{FSE}, Z_{FSI}, Z_{RSE}, Z_{RSI}$)
- (c) Pente (Z') à ces quatre points
- (d) Emplacement X_{SS} , hauteur (Z) et pente (Z') des deux points définis par la présence du longeron auxiliaire (Z_{SSE}, Z_{SSI} , optionnel)

5. Point d'inflexion sur l'intrados, (INF)

- (a) Position (X, Z) $_{INF}$ du point
- (b) Pente (Z') à ce point

6. Cavité, (CAV)

- (a) Position du point maximum de la cavité $(X, Z)_{CAV}$
- 7. Angle de torsion de la section par rapport à l'aile (α).
- 8. Points d'intersections du corps principal avec les becs et volets, (SE, SI, FE, FI)
 - (a) Position (X, Z) des 4 points définis par le dessus et le dessous du bec et du volet
 - (b) Pente (Z') à ces quatre points

Ces paramètres ne sont pas nécessairement tous présents en tout temps. Certains sont optionnels et d'autres dépendent de la stratégie de paramétrisation employée. Dans la paramétrisation PARSEC par exemple, seulement 11 paramètres sont utilisés (voir la section 1.1.4). Si nous utilisons une B-Spline, le nombre de paramètres employés peut varier et il dépend de la stratégie employée pour faire le passage entre les paramètres décrits ci-haut et la position des points de contrôle et des nœuds du vecteur nodal de la courbe.

Il faut donc trouver un lien entre certaines grandeurs géométriques et les variables de design brutes (coefficients PARSEC ou positions (poids) des points de contrôle pour les B-Splines (NURBS)) de la méthode de représentation. Il y a donc un prétraitement des contraintes pour construire le profil. Nous avons vu celui à appliquer aux paramètres PARSEC : deux systèmes linéaires sont solutionnés pour trouver les coefficients des fonctions polynomiales à partir de grandeurs géométriques. Dans le cas d'un profil défini par une courbe B-Spline, la transformation choisie dépend de la nature des contraintes à imposer. Dans le cas de la simple interpolation, des méthodes sont déjà développées (Piegl et Tiller 1995). Cependant, ici les contraintes sont des positions, mais aussi des pentes et des rayons de courbure. Deux méthodes d'interpolation avancées sont proposées au chapitre suivant pour répondre à ce besoin.

2.3 Les paramètres de la forme en plan

L'enveloppe extérieure de la forme en plan représente la projection normale de l'aile sur le sol et son aire définit la surface de référence. La forme en plan contient la position des principaux éléments de l'aile. Elle est constituée de plusieurs segments de droites et est définie à l'aide de la corde de trois sections : une à la base (emplanture), une à la cassure, et l'autre au bout (saumon) de l'aile. Le bord d'attaque est une ligne droite à un angle de flèche Λ par rapport à la direction d'avancement reliant les trois cordes à leur bord d'attaque. Le bord de fuite est en deux segments de droite : de la base à la cassure et de la cassure au bout (voir figure 2.3).

Nous retrouvons aussi sur la forme en plan la position des longerons avant et arrière ainsi que le longeron auxiliaire. Les longerons sont représentés par des segments de droites sur la forme en plan, car leur orientation perpendiculaire à l'écoulement fait que leurs projections se fondent en des droites. La position des longerons avant et arrière est déterminée en terme de pourcentage de corde sur les stations au bout et à la cassure. Une droite est tirée entre ces deux points puis étendue jusqu'à la base. Pour le longeron auxiliaire, il est maintenu parallèle au bord de fuite et passe par l'intersection de la section à la cassure avec le longeron arrière.

Enfin, nous voyons aussi les lignes représentant l'emplacement où les becs et volets se confondent avec le profil de croisière. La même stratégie de positionnement qu'avec les longerons est employée pour déterminer l'emplacement de ces lignes. Deux paramètres (cassure et bout) fixent chacune de ces quatre lignes. Voici une liste de tous les paramètres définissant la forme en plan :

1. Longueur de corde projetée à la base C_{base} , à la cassure $C_{cassure}$ et au bout C_{bout}
2. Demi-envergure $Y_{envergure}$ et emplacement de la cassure $Y_{cassure}$
3. Angle de flèche Λ du bord d'attaque

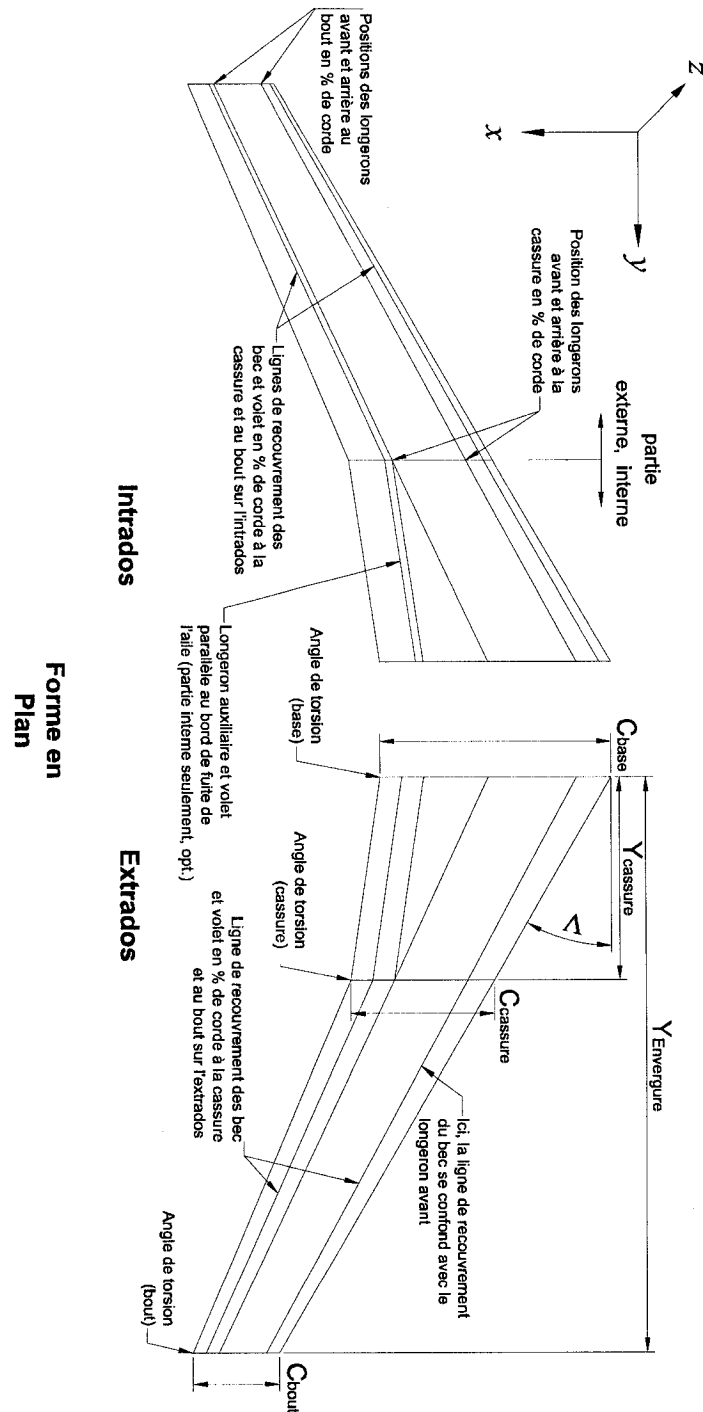


FIG. 2.3: Identification des paramètres géométriques pour définir la forme en plan

4. Emplacement (% corde) des longerons avant et arrière sur les sections à la cassure et au bout
5. Emplacement (% corde) de la ligne de recouvrement des volets avant et arrière sur l'extrados et l'intrados sur les sections à la cassure et au bout

2.4 Les caractéristiques de l'aile 3D

L'aile doit répondre aux exigences des deux sous-espaces 2D décrits ci-haut. Elle doit aussi satisfaire des critères plus globaux, comme l'absence d'oscillations de la surface dans le sens de l'envergure et être définie par des surfaces réglées en certains endroits.

Dans le cas des B-Splines, une façon de faire pour représenter la surface de l'aile, est d'ajouter un paramètre et un vecteur nodal. Nous obtenons alors une surface par produit tensorielle, *i.e.* deux directions paramétriques (u, v) . Elle s'écrit :

$$S(u, v) = \sum_{i=0}^n \sum_{j=0}^m N_{i,p}(u) N_{j,q}(v) \mathbf{P}_{i,j} \quad (2.2)$$

Cependant, l'utilisation de ce type de surface implique d'importantes concessions sur la flexibilité des vecteurs nodaux, et donc de la surface.

L'aile peut également être décrite comme la variation d'une section 2D en fonction de son envergure. Si chaque profil est décrit par un jeu de paramètre, la modélisation de l'aile se résume en l'ajout d'une dimension à ces paramètres. Cette stratégie s'avère utile pour utiliser la représentation PARSEC pour définir une aile, car cette paramétrisation n'est définie qu'en 2D. Elle peut aussi être facilement employée dans le cas où une B-Spline est utilisée pour représenter un profil 2D par le biais de paramètres d'ingénierie. Il s'agit de faire varier les paramètres en fonction de l'envergure. Nous devons alors utiliser une fonction par paramètre et développer une méthode pour les générer

automatiquement. En fonction des caractéristiques de la forme en plan, il paraît naturel d'appuyer ces fonctions sur les paramètres des profils à l'emplanture, à la cassure et au bout.

Les fonctions sélectionnées doivent être C^2 continues et varier le plus régulièrement possible. L'utilisation des coniques répond bien à cette tâche. Comme nous connaissons les caractéristiques générales aux trois profils guides, ces coniques devront interpoler les valeurs de paramètres correspondant à ces profils. La section 3.2 est dédiée à cette question.

Voici une liste des contraintes que l'aile doit respecter :

1. La surface de l'aile est continue C^2 (continuité de courbure) en tous points sauf à la cassure. Une continuité inférieure (C^1 et moins) donne de piètres performances aérodynamiques et une supérieure (C^3 et plus) peut entraîner des oscillations en plus d'être plus difficile à générer.
2. Les becs et volets sont définis par des surfaces réglées. Ici les becs et volets sont définis comme étant les trois surfaces comprises :
 - (a) entre le bord de fuite et la ligne de recouvrement arrière sur l'extrados
 - (b) entre le bord de fuite et la ligne de recouvrement arrière sur l'intrados
 - (c) entre les lignes de recouvrement avant en passant par le bord d'attaque
3. La courbe de l'épaisseur dans le sens de l'envergure doit s'approcher d'une conique dans la partie interne et d'une droite dans la partie externe
4. Transformation bijective (*mapping*) unique entre les paramètres et la géométrie

Faisons une mise au point à propos de la torsion de l'aile. Les cordes des profils constituant les stations de l'aile ne sont généralement pas parallèles à l'écoulement. Ils ont un angle d'incidence inclus par construction, ou angle de torsion. La variation de cet angle dans le sens de l'envergure est appelée distribution de torsion. Usuellement, cet angle est de quelques degrés à l'emplanture et est légèrement négatif au bout de l'aile.

Généralement, les stations sont tournées autour du point situé au quart de la corde, de sorte qu'après une rotation d'un angle α positif le bord d'attaque est déplacé vers le haut et le bord de fuite vers le bas. Comme ce sont des droites qui relient ces points dans le sens de l'envergure, la distribution de torsion est donnée par les trois angles d'incidence aux stations guides, comme le montre la figure 2.4. La non-linéarité vient de la fonction tangente qui représente la distribution de torsion.

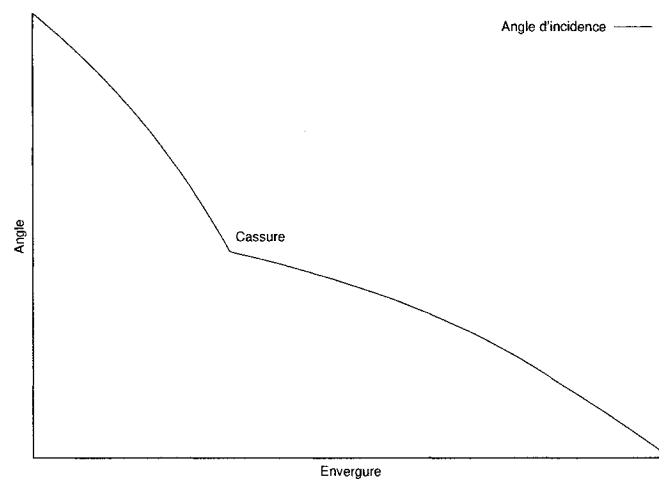


FIG. 2.4: Distribution de torsion

En résumé, nous voulons construire de façon automatisée une aile transsonique tridimensionnelle. Cette aile est entièrement définie géométriquement à l'aide d'une série de paramètres d'ingénierie référant aux trois profils guides et à la forme en plan. À partir de ceux-ci, un squelette tridimensionnel de caractéristiques géométriques est construit sur lequel va s'appuyer la surface extérieure de l'aile.

2.5 Logiciel

Le code devra pouvoir construire de façon autonome une aile à l'aide de fichiers de données. Comme des programmes d'optimisation vont, pour effectuer leur tâche, modifier les fichiers d'entrée et lancer le module de construction un grand nombre de fois, celui-ci

doit être robuste. Nous avons choisi de concevoir ce programme à l'aide des librairies *Pirate* et *opgeom*¹. Les raisons sont multiples. Dans ces librairies, les principales méthodes de représentation et manipulation de courbes et surfaces B-Spline et NURBS sont implémentées. Ces librairies s'appuient sur un format d'échange de donnée, le fichier *Pirate Import-Export*, qui permet l'écriture et la lecture d'entités géométriques et topologiques en plus de maillages et de solutions. Elle propose aussi un patron générique d'application à partir duquel il est aisé d'interagir avec les objets des librairies. Enfin, notons la disponibilité de personnes ressources liées à ces librairies.

Le format et le nombre des fichiers d'entrée ne sont pas figés définitivement. Une première version a été développée, mais le format utilisé n'est peut-être pas final. Donc, toutes les entrées doivent demeurer souples. En effet, les données devront éventuellement se conformer au format développé en tâche 1 du projet MOSAIC dans lequel ce travail s'inscrit. (voir tableau 1 en page 4). Il est donc naturel de regrouper les informations de même type dans des fichiers distincts. Ici, nous utilisons trois fichiers de paramètres. Le premier contient les informations relatives à la construction et à la discrétisation de l'aile. Le second est constitué des informations relatives à la forme en plan. Enfin, le dernier contient des paramètres d'ingénierie adimensionnels aux trois profils guides de l'aile. Voir l'annexe III pour plus de détails.

Tout comme pour les fichiers d'entrée, le format des fichiers de sortie n'est pas figé. Il est naturel de stocker le modèle sous le format propriétaire des librairies utilisées, *i.e.* le format *pirate* (fichier avec l'extension *.PIE* - *Pirate Import-Export*). Il faut aussi garder à l'esprit la possibilité d'exporter la géométrie sous forme de nuages de points.

Une approche orientée objet est privilégiée. Nous pouvons ainsi facilement dériver de nouveaux objets, pour de nouvelles applications. Les objets et méthodes développés vont être utilisables et exportables dans le cadre de développements futurs et d'utilisation

¹www.polymtl.ca/grmio/magnu/

à d'autres fins. De plus, en procédant ainsi, l'expertise accumulée peut se gérer plus aisément.

CHAPITRE 3

COURBES CONTRAINTES

PARMI les requis mentionnés au chapitre 2, le contrôle des caractéristiques géométriques à certains endroits sur des courbes apparaît comme l'un des plus importants. Nous entendons par caractéristiques géométriques les positions, tangentes et courbures cibles. Ce chapitre traitera donc de quelques méthodes pour imposer des contraintes efficacement sur des courbes lors de leur construction. Il sera question dans un premier temps, à la section 3.1, de contraindre une B-Spline. Deux approches différentes y seront étudiées. La première construit la B-Spline résultante en joignant ensemble une série de B-Splines contraintes. La seconde propose une méthode d'interpolation avancée permettant la construction d'une B-Spline respectant un nombre variable de contraintes de différentes natures. Ces deux méthodes seront comparées afin de déterminer laquelle convient le mieux pour notre application. Dans un deuxième temps, à la section 3.2, les coniques seront étudiées plus en détail. Cette grande famille de courbes va être adaptée afin de répondre à nos besoins pour l'interpolation des paramètres dans le sens de l'envergure (voir section 2.4 et chapitre 4).

3.1 B-Splines contraintes

Dans cette section, nous étudions en détail deux méthodes d'imposition de contraintes géométriques sur des B-Splines. La courbe résultante devra passer par une série de points où, en plus de la position, pourront y être imposées l'angle de la pente, la courbure et/ou l'amplitude de la dérivée première. La première méthode n'impose les contraintes qu'aux extrémités d'une B-Spline, deux points à la fois. Un collage est donc requis

pour obtenir la courbe finale. La seconde méthode, inspirée de méthodes utilisées dans certains logiciels de design interactif ainsi que d'algorithmes d'interpolation de base, peut imposer les contraintes des différents points en une étape globale. Dans les deux méthodes, seul le cas bidimensionnel est étudié. Ces études pourraient être généralisées au cas tridimensionnel, mais comme l'objectif ici est de représenter des stations (profils bidimensionnels) se confondant sur un plan, ce cas particulier est suffisant.

Nous amenons maintenant la nomenclature *point clé* pour désigner un point où des contraintes géométriques sont spécifiées. Un point clé, noté \mathbf{Q}_i^f , est caractérisé par son ordre f et son indice i . Un ordre zéro (0) signifie que seule la position est donnée à ce point ; un ordre de un (1) indique que l'angle de la pente est aussi contraint ; enfin, un ordre de deux (2) indique que la courbure est également contrainte. L'angle de la pente est donné en degré pour un maximum de lisibilité et doit être compris entre 0° et 360° . L'avantage d'utiliser un angle défini de la sorte contrairement à la pente, est la plus grande précision. D'abord, nous évitons les problèmes numériques de la manipulation de l'infini ($m = \infty$ vs $\angle = 90^\circ$). Ensuite, nous distinguons également la direction avec l'angle, c'est la différence entre 90° et -90° , contrairement à $m = \infty$, qui n'apporte pas cette précision. Notons aussi qu'à partir du moment où l'ordre est supérieur ou égal à 1, l'amplitude de la dérivée première peut aussi être contrainte. Des explications détaillées sur cette grandeur sont fournies à la section 3.1.1.1. Si l'amplitude n'est pas spécifiée, le système en impose une automatiquement. Il y a donc un maximum de cinq valeurs pouvant être contraintes par point clé : la position ($\mathbf{Q}_i^{0x}, \mathbf{Q}_i^{0y}$), l'angle de la pente ($\mathbf{Q}_i^{1\angle}$), la courbure ($\mathbf{Q}_i^{2\kappa}$), et l'amplitude de la dérivée première ($\mathbf{Q}_i^{1\omega}$). L'indice i sert à différencier les points clés et $0 \leq i < l$, où l est le nombre de points clés.

3.1.1 B-Spline contrainte aux extrémités

Dans un premier temps, la méthode d'obtention d'une courbe ayant des caractéristiques géométriques (position, angle de la pente, courbure) prescrites en plusieurs points clés est subdivisée en parties. Ainsi, une B-Spline est construite entre deux points clés consécutifs. Cette opération est répétée pour toutes les paires de points clés consécutifs. Donc, pour l points clés, il faut construire $l - 1$ B-Splines. Lorsque nous avons l'ensemble des B-Splines constituant la géométrie, nous pouvons les rabouter pour qu'elles ne forment plus qu'une seule courbe. Celle-ci satisfera toutes les contraintes désirées. Dans cette méthode, l'ordre de tous les points clés doit être de $f = 2$.

Le principal objectif de cette section est de présenter une méthode permettant de contraindre une B-Spline à ses deux extrémités. Comme ces B-Splines n'ont qu'un point clé au début et à la fin, l'imposition de contraintes ne se fait qu'aux paramètres d'avancement $u = 0$ et $u = 1$. Nous avons donc 6 contraintes en tout : la position, l'angle de la pente et la courbure aux deux extrémités, comme l'illustre la figure 3.1. Comme le degré le plus élevé des contraintes est 2 (courbure), notre courbe doit avoir un degré $p = 3$ pour assurer la continuité sur la courbe.

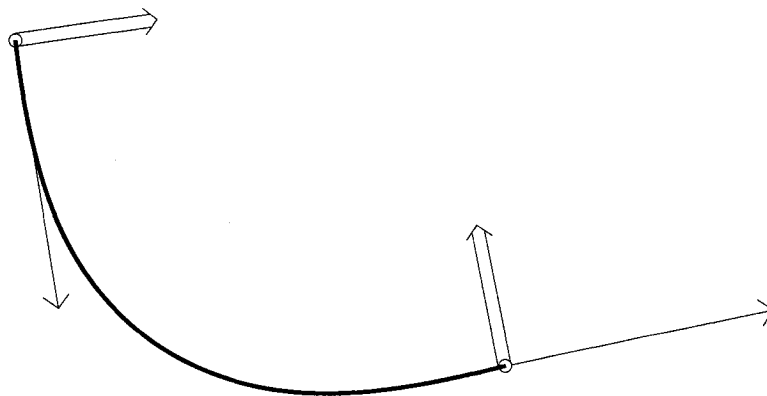


FIG. 3.1: Représentation des différents niveaux de contraintes aux extrémité sur une B-Spline (Position : ○, Pente : → et Courbure : ⇒)

Pour arriver à imposer ces conditions, il nous faut en premier des expressions de la posi-

tion des points de contrôle en fonction des dérivées aux extrémités, ce que nous fournit Piegl et Tiller (1995). Ensuite, nous devons développer une méthode pour construire la courbe à partir des contraintes. Les principaux défis sont de prétraiter les contraintes afin d'établir une bonne relation entre celles-ci et les valeurs de dérivées, et d'obtenir un bon vecteur nodal à partir des contraintes. Revenons d'abord sur la définition des B-Splines (équation 1.23) et de leurs dérivées :

$$\mathbf{C}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i \quad (3.1)$$

$$\mathbf{C}'(u) = \sum_{i=0}^n N'_{i,p}(u) \mathbf{P}_i \quad (3.2)$$

où \mathbf{P}_i sont les $n + 1$ points de contrôle et $N_{i,p}$ sont les $n + 1$ fonctions de bases de degré p associées aux points de contrôle. Nous savons (voir annexe I), que de façon générale et récurrentes, les dérivées d'ordre k peuvent être exprimées par :

$$\mathbf{C}^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p-k}(u) \mathbf{P}_i^{(k)} \quad (3.3)$$

Avec

$$\mathbf{P}_i^{(k)} = \begin{cases} \mathbf{P}_i & \text{si } k = 0 \\ \frac{p - k + 1}{u_{i+p+1} - u_{i+k}} (\mathbf{P}_{i+1}^{(k)} - \mathbf{P}_i^{(k)}) & \text{si } k > 1 \end{cases} \quad (3.4)$$

La courbe représentant la $k^{\text{ième}}$ dérivée, est une nouvelle B-Spline de degré $p - k$ qui utilise de nouveaux points de contrôle, issus d'une interpolation entre les points originaux. Cette notation permet de faire un lien direct entre la position des points de contrôle et les contraintes à respecter.

À l'aide des équations 3.3 et 3.4, la dérivée première ($k = 1$) de la courbe s'exprime aux

deux extrémités ($u_0 = 0$ & $u_m = 1$, voir annexe I) comme suit ($n = 5$) :

$$\mathbf{C}'(0) = \mathbf{P}_0^{(1)} = \frac{3}{u_4} (\mathbf{P}_1 - \mathbf{P}_0) \quad (3.5)$$

$$\mathbf{C}'(1) = \mathbf{P}_4^{(1)} = \frac{3}{1 - u_5} (\mathbf{P}_5 - \mathbf{P}_4) \quad (3.6)$$

En répétant le même travail avec $k = 2$ pour exprimer la dérivée seconde aux deux extrémités, on obtient :

$$\mathbf{C}^{(2)}(0) = \mathbf{P}_0^{(2)} = \frac{6}{u_4} \left(\frac{\mathbf{P}_0}{u_4} - \frac{(u_4 + u_5)\mathbf{P}_1}{u_4 u_5} + \frac{\mathbf{P}_2}{u_5} \right) \quad (3.7)$$

$$\mathbf{C}^{(2)}(1) = \mathbf{P}_3^{(2)} = \frac{6}{1 - u_5} \left(\frac{\mathbf{P}_5}{1 - u_5} - \frac{(2 - u_5 - u_4)\mathbf{P}_4}{(1 - u_5)(1 - u_4)} + \frac{\mathbf{P}_3}{1 - u_4} \right) \quad (3.8)$$

Nous pouvons tirer des expressions 3.5 à 3.8 quelques remarques intéressantes. D'abord, notons que deux points de contrôle influencent la dérivée première aux extrémités et trois pour la dérivée seconde. Comme les B-Splines interpolent le premier et dernier point de contrôle, ceux-ci sont déterminés par la position des points clés.

En ce qui concerne la dérivée première aux extrémités, deux points de contrôle sont nécessaires pour l'imposer. Les points de contrôle aux extrémités étant déjà fixés par la position, il n'en reste qu'un à déterminer pour contraindre la dérivée première. Nous y arrivons en isolant \mathbf{P}_1 et \mathbf{P}_4 des expressions 3.5 et 3.6. Les dérivées premières aux extrémités fixent les positions du deuxième et avant-dernier point de contrôle.

Due à la nature récursive de la formulation, le même constat est fait avec la dérivée seconde. Celle-ci fait intervenir trois points de contrôle, mais comme deux sont déjà fixés, le dernier est suffisant pour imposer la dérivée seconde. En somme, trois points de contrôle par extrémité sont nécessaires pour imposer des conditions de position, tangence et courbure. Les B-Splines reliant deux points clés devront donc avoir six points de contrôle pour se plier aux exigences. Notre système est cohérent, car nous avons au-

tant d'inconnues (position des points de contrôle) que d'équations (contraintes géométriques). Comme le degré est fixé ($p = 3$) tout comme le nombre de points de contrôle (6 points $\rightarrow n = 5$), il en résulte que le nombre de nœuds du vecteur nodal l'est aussi, $m = n + p + 1 = 9$. Comme nous employons un vecteur nodal ouvert, les quatre premiers (de 0 à 3) et les quatre derniers (de 6 à 9) nœuds sont déterminés (respectivement 0 et 1), il ne reste plus que deux nœuds internes à trouver, u_4 et u_5 . Nous suggérons quelques méthodes pour y parvenir à la section 3.1.1.2.

3.1.1.1 De contraintes à dérivées

Nous avons à présent les relations fixant la position des points de contrôles en fonction des dérivées fournies. Avant d'en arriver à la B-Spline contrainte, il faut d'abord des relations entre les contraintes géométriques données et les dérivées, ce que présente cette section, et le vecteur nodal. C'est de cette étape de prétraitement que provient l'amplitude ($\mathbf{Q}_i^{l'c}$) dont nous parlions plus haut. L'expression paramétrique générale liant la courbure aux dérivées première et seconde d'une courbe est donné par :

$$\kappa = \frac{1}{\rho} = \frac{x'y'' - y'x''}{(x'^2 + y'^2)^{3/2}} \quad (3.9)$$

À l'aide de l'équation 3.9, nous pouvons isoler la dérivée seconde étant donné que les autres variables sont connues. Remarquons qu'à ce moment, pour chaque ordre de dérivées, nous doublons les variables, car il y en a une par composante. Comme nous travaillons avec des vecteurs de dimension 2, il doit aussi y avoir deux contraintes par ordre de dérivée. Bien que la position, déjà fournie sous forme vectorielle (x et y), s'accorde bien de ce constat, ce n'est pas le cas pour l'angle de la pente et la courbure, qui sont deux scalaires.

Il y a donc, pour chaque extrémité de la courbe plus d'inconnues que d'équations. En effet, en tenant compte des deux composantes, nous n'avons que quatre contraintes — la position (Q_i^{0x}, Q_i^{0y}) , l'angle de la pente (Q_i^{1x}) et la courbure (Q_i^{2x}) — pour les six inconnus (x, y, x', y', x'', y'') . Nous devons trouver un moyen de fermer ce système.

Pour la position, la solution est triviale. Les composantes de la coordonnée du point clé fixent les deux premières inconnues, x et y . Il nous reste deux contraintes pour déterminer quatre inconnues.

En ce qui concerne la pente, nous avons $\frac{dy}{dx}$. Utilisant la différenciation en chaîne, nous pouvons exprimer

$$\frac{dy}{dx} = \frac{dy/du}{dx/du} = \frac{y'}{x'} \quad (3.10)$$

Les termes x' et y' , qui représentent la dérivée d'une composante selon u , peuvent être interprétés comme les composantes de la vitesse d'avancement instantanée sur la courbe. L'angle de la pente que nous avons comme contrainte représente alors la direction de cette vitesse, sa norme (ou amplitude) étant inconnue. Figuré sous cet angle, l'amplitude est une nouvelle variable de design qui modifie la grandeur de la région d'influence des contraintes, comme le montre la figure 3.2. Sur celle-ci, l'amplitude au début d'une courbe (le début est à l'origine) est augmentée puis diminuée de 50%. On remarque que les deux courbes modifiées ne sont plus symétriques et que l'amplitude agit sur la grandeur de la zone près du point clé.

L'amplitude peut être fixée de façon interne automatiquement ou alors l'utilisateur peut la spécifier pour avoir davantage de contrôle sur la courbe. Nous allons voir la stratégie adoptée dans chacun des cas. Si nous voulons que l'amplitude soit ajustée automatiquement, nous devons trouver un moyen de la fixer qui donne une courbe où la paramétrisation de la courbe est le plus près possible d'une paramétrisation intrinsèque, *i.e.* une paramétrisation ayant une vitesse de parcours constante. Une bonne façon de faire consiste à fixer la norme du vecteur vitesse aux extrémités égale à la longueur de

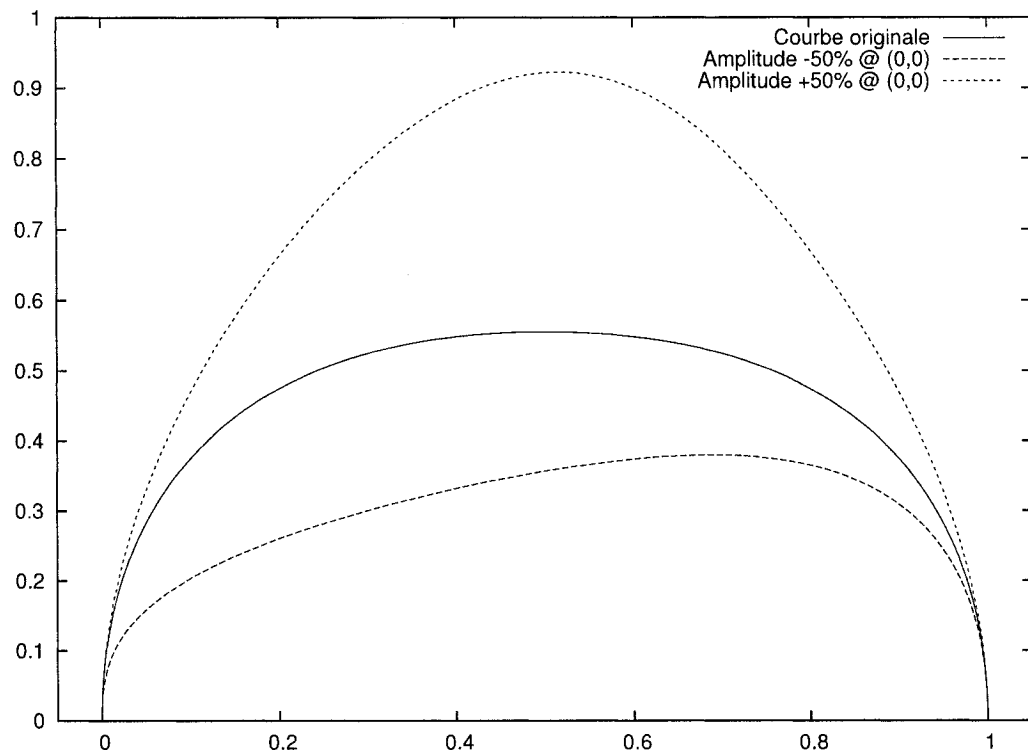


FIG. 3.2: Effet de l'amplitude de la dérivée au début d'une courbe (origine)

la courbe. Comme cette dernière quantité n'est pas connue *a priori*, nous devons itérer sur celle-ci. Une première évaluation de la longueur est obtenue par la grandeur du segment reliant les deux points clés. Une courbe contrainte est générée à partir de ces caractéristiques. La longueur de cette nouvelle courbe est extraite, puis comparé à celle de l'itération précédente. Si l'erreur commise sur la longueur est plus petite qu'une tolérance donnée, la dernière courbe construite est considérée comme la bonne. Par contre, si l'erreur est trop grande, une nouvelle courbe est générée en prenant bien soin d'actualiser l'amplitude avec la longueur de la courbe actuelle. Généralement, moins de 6 itérations sont nécessaires pour converger vers des niveaux d'erreurs sur la longueur de l'ordre de 10^{-3} , ce qui est jugé satisfaisant.

L'amplitude peut aussi être fournie par l'utilisateur. Dans ce cas elle peut être vue de façon dimensionnelle ou adimensionnelle (en agissant comme un facteur multiplicateur

sur la corde de référence pour ajuster l'amplitude). Dans le premier cas, l'amplitude fournie est stockée en mémoire et la courbe recherchée est construite en une seule étape, car il n'y a plus besoin d'itérer sur la longueur de la courbe. Cependant, il est préférable de la donner sous forme adimensionnelle, car les manipulations s'en trouvent simplifiées et la signification de ce paramètre est davantage intuitive. Si elle est fournie de façon adimensionnelle, la même stratégie que celle utilisée dans le cas où le système impose l'amplitude est employée. La seule différence est que la longueur est multipliée par le facteur adimensionnel pour actualiser l'amplitude au lieu d'être utilisée telle quelle. C'est cette seconde technique qui a été retenue, car des essais préliminaires ont montré que cette technique est plus robuste et permet de construire de meilleures courbes lorsque les points clés sont distribués irrégulièrement. De plus elle a l'avantage de s'actualiser automatiquement si la position d'un point clé est changée.

En ce qui concerne la courbure et donc, les dérivées secondes, les choix sont plus faciles à faire. Nous voulons imposer la courbure qui est reliée par l'équation 3.9 aux dérivées. Comme la courbure est donnée et que les dérivées premières sont déjà fixées, il ne reste que les dérivées secondes pour agir sur celle-ci. Pour les dérivées secondes, contrairement aux dérivées premières, nous n'avons de l'information que sur sa norme, la direction restant à fixer. Un choix logique pour ce faire, dans le but d'avoir les courbes les plus régulières possibles, est d'imposer cette direction perpendiculaire à la tangente à ce même point. Cette nouvelle condition donne aux courbes une belle allure (variation de la courbure plus régulière de part et d'autre du point clé) et l'effet des paramètres est plus intuitif. Nous venons ainsi de fixer les inconnues restantes du système. Mis en équation, les relations entre contraintes (Position $(\mathbf{Q}_i^{0x}, \mathbf{Q}_i^{0y})$, Angle $(\mathbf{Q}_i^{1\angle})$, Courbure $(\mathbf{Q}_i^{2\kappa})$, Amplitude $(\mathbf{Q}_i^{1\omega})$) et dérivée sont :

$$\begin{cases} x &= \mathbf{Q}_i^{0x} \\ y &= \mathbf{Q}_i^{0y} \end{cases} \quad (3.11)$$

$$\begin{cases} x' &= LQ_i^{1\leftrightarrow} \cos(Q_i^{1\leftrightarrow}) \\ y' &= LQ_i^{1\leftrightarrow} \sin(Q_i^{1\leftrightarrow}) \end{cases} \quad (3.12)$$

$$\begin{cases} x'' &= -y' Q_i^{2\kappa} \sqrt{x'^2 + y'^2} \\ y'' &= x' Q_i^{2\kappa} \sqrt{x'^2 + y'^2} \end{cases} \quad (3.13)$$

où L est la longueur de la courbe, la valeur sur laquelle il faut itérer. La dernière étape avant de pouvoir générer définitivement la courbe, est une méthode d'obtention du vecteur nodal que l'on présente dans la section suivante.

3.1.1.2 Méthode de construction du vecteur nodal

Il ne reste plus qu'à faire le vecteur nodal pour générer la B-Spline contrainte. Pour le construire, rappelons d'abord que nous cherchons une B-Spline avec 6 points de contrôle, donc $n = 5$ (\mathbf{P}_0 à $\mathbf{P}_5 \rightarrow 6$ pts). Aussi, comme une continuité C2 est recherchée, une courbe de troisième degré a été sélectionnée, alors $p = 3$. Comme le nombre d'éléments du vecteur nodal est $m = n + p + 1 = 9$ (de 0 à 9 = 10 nœuds) et que les $p + 1$ premiers et derniers nœuds sont connus (voir eq. I.2), il ne reste que deux éléments internes à déterminer, les nœuds u_4 et u_5 .

Élaborons maintenant des méthodes de construction du vecteur nodal, ce qui est loin d'être trivial étant donné toute la subtilité des choix qui s'offrent à nous. En effet, il n'existe pas une méthode mathématique unique pour construire le vecteur nodal. Même pour le cas le plus simple, l'interpolation de points par une B-Spline, la construction du vecteur nodal est une étape cruciale. Plusieurs méthodes ont été développées (uniforme, longueur de corde, centripète et logarithmique) et chacune d'entre elles a ses avantages et inconvénients. C'est davantage sur une base empirique que telle ou telle méthode est préconisée. L'ensemble de ces techniques utilise la distance entre les points comme information pour construire le vecteur nodal. Ici, nous avons un problème d'interpolation

avec pente (dérivée première) et rayon de courbure (dérivée seconde) : il s'agit donc d'une extension du problème d'interpolation classique. Alors, comme dans ce dernier cas, il ne faut pas s'attendre à avoir une méthode parfaite, mais plutôt plusieurs façons de faire ayant chacune ses caractéristiques propres. Ces différentes méthodes peuvent inclure, en plus de la distance séparant les points, de l'information provenant de la pente et de la courbure. Les méthodes que nous proposons sont sommairement décrites ci-dessous. Nous pouvons aussi voir leurs effets sur la figure 3.3.

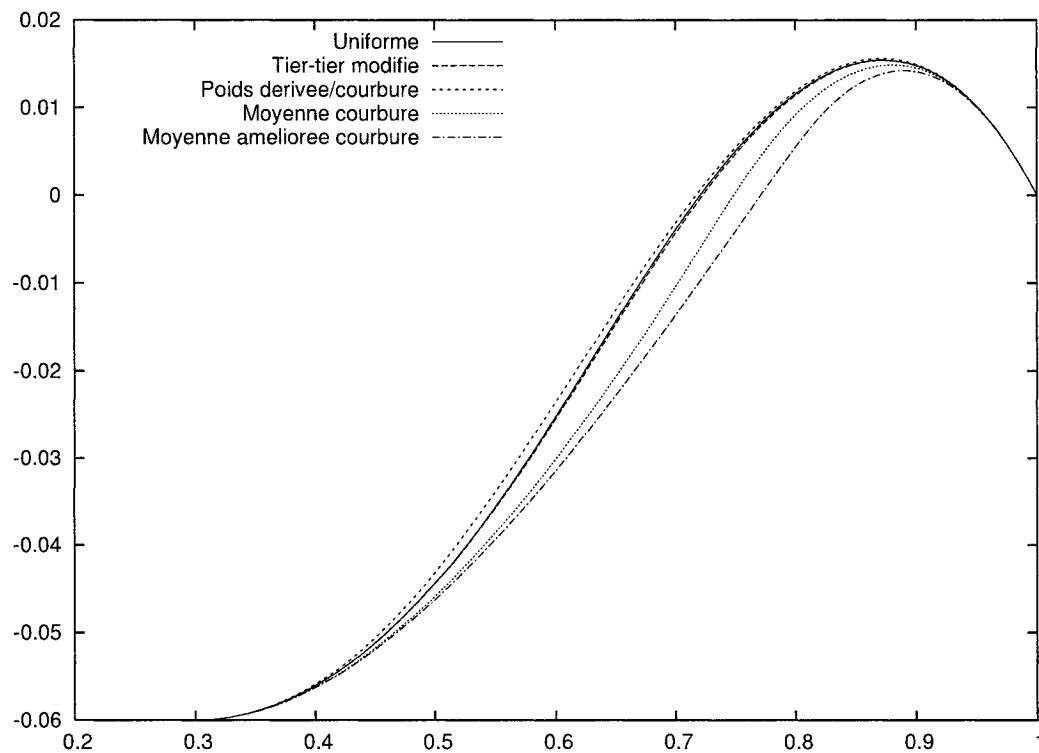


FIG. 3.3: Courbes interpolant les points $(0,3,-0.06)$ à 0° , $crb=0,8$ et $(1,0)$ à -15° , $crb=-2,5$ avec différents vecteurs nodaux. Cette courbe est caractéristique d'un intrados de profil transsonique.

Uniforme : Cette méthode est la plus simple, mais les essais préliminaires ont montré qu'elle donne en contrepartie les moins bons résultats. Elle ne tient pas compte de la géométrie à construire. Le vecteur nodal est simplement divisée en intervalle régulier, donnant $u_4 = 1/3$ et $u_5 = 2/3$.

Tier-tier modifié : Cette méthode s'inspire de la précédente, en ce sens qu'elle positionne les nœuds uniformément si des valeurs nulles d'amplitude et de courbure sont saisies. Ces nœuds sont donc déplacés en fonction de la courbure (\mathbf{Q}_i^{2k}), de la norme de la dérivée ($\mathbf{Q}_i^{1\leftrightarrow}$) et d'un paramètre de décalage ($D=10$). L'expression pour les nœuds internes est :

$$\begin{cases} u_4 = \frac{1}{3} \left(\frac{1 + \mathbf{Q}_i^{1\leftrightarrow}/D}{1 + |\mathbf{Q}_i^{2k}|/L/D} \right) \\ u_5 = 1 - \frac{1}{3} \left(\frac{1 + \mathbf{Q}_{i+1}^{1\leftrightarrow}/D}{1 + |\mathbf{Q}_{i+1}^{2k}|/L/D} \right) \end{cases} \quad (3.14)$$

Si nous prenons exemple sur la première extrémité ($u = 0$), plus l'amplitude est grande, plus loin est le nœud, tandis qu'une grande courbure le rapproche. Ce comportement est désiré afin d'amenuiser les variations de courbure sur la courbe résultante.

Poids dérivée/courbure par moitié : Cette méthode prend en compte les courbures et amplitudes des dérivées comme la précédente, mais alloue la moitié de l'espace paramétrique à chaque extrémité. La valeur des nœuds est obtenue par moyennage avec l'amplitude de la dérivée et par division par la courbure. L'expression pour les nœuds internes est :

$$\begin{cases} u_4 = \frac{1}{2} \frac{L\mathbf{Q}_i^{1\leftrightarrow}}{L\mathbf{Q}_i^{1\leftrightarrow} + |\mathbf{Q}_i^{2k}|} \\ u_5 = 1 - \frac{1}{2} \frac{L\mathbf{Q}_{i+1}^{1\leftrightarrow}}{L\mathbf{Q}_{i+1}^{1\leftrightarrow} + |\mathbf{Q}_{i+1}^{2k}|} \end{cases} \quad (3.15)$$

Moyenne des courbures : Cette méthode, ne se base que sur les courbures. En fait, elle prend en compte les courbures aux extrémités, mais aussi la variation de l'angle que la courbe subit lors de son parcours, qui est aussi approximativement une courbure. Un ratio de ces valeurs est calculé pour obtenir la position des nœuds.

Nous devons d'abord calculer deux valeurs intermédiaires :

$$\Sigma = \frac{1}{1 + |\mathbf{Q}_i^{2k}|} + \frac{1}{1 + \frac{\Delta\theta}{\Delta L}} + \frac{1}{1 + |\mathbf{Q}_{i+1}^{2k}|} \quad (3.16)$$

où

$$\frac{\Delta\theta}{\Delta L} = \frac{|\mathbf{Q}_i^{1k} - \mathbf{Q}_{i+1}^{1k}|}{L} \quad (3.17)$$

L'expression pour les nœuds internes est :

$$\begin{cases} u_4 = \left(\frac{1}{1 + |\mathbf{Q}_i^{2k}|} \right) / \Sigma \\ u_5 = 1 - \left(\frac{1}{1 + |\mathbf{Q}_{i+1}^{2k}|} \right) / \Sigma \end{cases} \quad (3.18)$$

Moyenne améliorée des courbures : Cette méthode est semblable à la précédente. La grande amélioration est que tout est calculé de façon adimensionnelle. De cette façon, appliquer une transformation affine sur les points clés, donne le même résultat que l'appliquer aux points de contrôles résultants. Cette propriété n'est pas toujours présente dans les méthodes énoncées plus haut.

Une autre différence vient du calcul de la variation de la pente. Au lieu de comparer les deux valeurs extrêmes, nous les comparons par rapport au vecteur direction reliant le premier point clé au deuxième. Les résultats sont similaires au cas précédent, sauf dans le cas où un point d'inflexion se trouve sur la courbe, un cas fréquent dans nos modélisations. Nous devons ici encore dans un premier temps calculer certaines valeurs intermédiaires :

$$\Sigma = \frac{1}{1 + L|\mathbf{Q}_i^{2k}|} + \frac{1}{1 + \delta\theta} + \frac{1}{1 + L|\mathbf{Q}_{i+1}^{2k}|} \quad (3.19)$$

où

$$\delta\theta = \delta_0 + \delta_1 \quad (3.20)$$

et

$$\begin{cases} \delta_0 = |\mathbf{Q}_i^{1^k} - \gamma| \\ \delta_1 = |\mathbf{Q}_{i+1}^{1^k} - \gamma| \end{cases} \quad (3.21)$$

Enfin

$$\gamma = \arctan\left(\frac{\mathbf{Q}_{i+1}^{0^y} - \mathbf{Q}_i^{0^y}}{\mathbf{Q}_{i+1}^{0^x} - \mathbf{Q}_i^{0^x}}\right) \quad (3.22)$$

L'expression pour les nœuds internes est donné par :

$$\begin{cases} u_4 = \left(\frac{1}{1 + L|\mathbf{Q}_i^{2^k}|}\right)/\Sigma \\ u_5 = 1 - \left(\frac{1}{1 + L|\mathbf{Q}_{i+1}^{2^k}|}\right)/\Sigma \end{cases} \quad (3.23)$$

Cette méthode donne de meilleurs résultats, car ici les valeurs sur lesquelles les ratios se basent sont sans dimension.

De nouvelles méthodes peuvent encore être développées. Notons que les différences entre les méthodes sont faibles. Cependant, il a été remarqué que la méthode uniforme a tendance à diverger sur la longueur lorsque de trop grandes variations de courbure sont présentes. Comme la dernière méthode (moyenne améliorée des courbures) semble très bien se comporter même lorsque de fortes variations sont imposées, celle-ci sera retenue comme méthode par défaut lorsque l'utilisateur n'en spécifie pas.

Ces méthodes ont été développées sur l'expérience empirique accumulée à force de manipuler des courbes dans différentes situations. Des cas test inventés ou provenant de la définition géométrique de profils ont été développés afin de comparer les méthodes en de multiples circonstances. L'évolution de la courbure, de l'amplitude et les variations de pente sur la courbe ont servi comme critères pour déterminer si une méthode de construction du vecteur nodal était valide. Ces critères ont également servi de balises dans le but d'améliorer ou de développer de nouvelles méthodes. La figure 3.4 montre que toutes les méthodes présentées donnent toutes approximativement une paramétri-

sation intrinsèque, car les fonctions $f(u) = \frac{L_i(u)}{L_{tot}}$ sont linéaires. Ce processus itératif a abouti sur la méthode de moyenne améliorée des courbures pour construire le vecteur nodal.

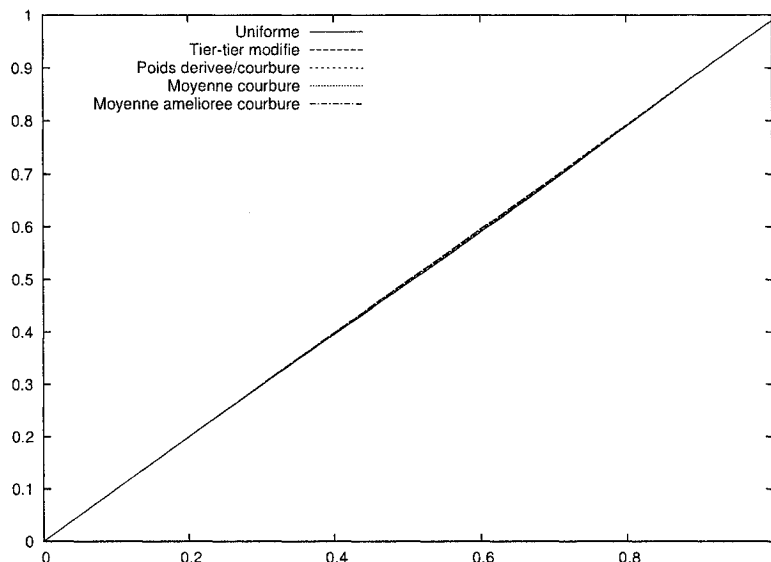


FIG. 3.4: Longueur normalisée de courbes en fonction de l'avancement paramétrique u sur le même cas qu'à la figure 3.3

3.1.1.3 Collage final des courbes

Nous avons à présent une liste des courbes reliant chacune deux points clés. Il faut maintenant les joindre entre elles afin de constituer une seule courbe finale. Pour ce faire, nous allons utiliser les algorithmes de manipulations de courbes de Piegl et Tiller (1995). Cependant, avant de faire le collage, nous normalisons les vecteurs nodaux de chaque courbe par sa longueur. L'idée est de conserver le plus possible la paramétrisation intrinsèque que nous tentons de donner à la courbe finale. Pour ce faire, nous multiplions tous les nœuds de chaque courbe primaire par sa longueur L_i de sorte que l'intervalle paramétrique passe de $0 \leq u \leq 1$ à $0 \leq \bar{u} \leq L_i$. En procédant ainsi, l'avancement paramétrique sur la courbe finale engendré par une courbe primaire est proportionnel à sa longueur. Donc, une petite courbe primaire engendrera un petit avancement paramétrique \bar{u} , car sa

longueur n'aura pas un grand poids par rapport à la longueur de la courbe finale. Une grande courbe aura l'effet contraire, *i.e.* un grand avancement paramétrique \bar{u} .

Une fois le collage effectué, nous divisons la valeur de chaque nœud par la valeur du dernier nœud de façon à normaliser la courbe finale dans un intervalle paramétrique standard ($u \in [0, 1]$). De plus, il est possible d'éliminer un nœud par jonction. Il reste donc des nœuds de multiplicité $s = 3$ après jonction. Nous ne pouvons pas descendre à une multiplicité de $s = 1$, car l'amplitude de la dérivée première est discontinue. Cependant, cette discontinuité ne gêne pas la géométrie, car la pente et le rayon de courbure sont continus par construction. Ceci est vérifié en constatant que trois points de contrôle sont disposés linéairement près de chaque jonction, comme le montre la figure 3.5. En effet, la distance séparant les deux points de contrôle sur les bord entre le central n'est pas la même, ce qui implique un saut dans la dérivée première. Cependant, la discontinuité n'est que de nature mathématique (C2 vs G2, continuité géométrique suffisante), car la courbure à ce point est la même et varie de façon continue de part et d'autre de ce point.

3.1.1.4 Diminution de l'ordre des points clés

Avec les développements faits dans cette section, nous avons maintenant les outils pour fabriquer une B-Spline cubique avec six points de contrôle sur laquelle les positions, pentes et courbures sont spécifiées. Afin de gagner en flexibilité et en souplesse d'utilisation, nous devons développer un moyen permettant l'interpolation même si la courbure n'est pas connue en un ou plusieurs points clés. Cette situation est souvent rencontrée dans le cadre de ce travail, car généralement, la courbure n'est disponible qu'en quelques points. Nous voulons que les conditions sur l'ordre soient $f \geq 1$ au lieu de $f = 2$ pour tous les points clés \mathbf{Q}_i^f . Cependant, au minimum un point clé doit avoir un ordre de $f = 2$, ce qui est toujours rencontré au bord d'attaque du profil.

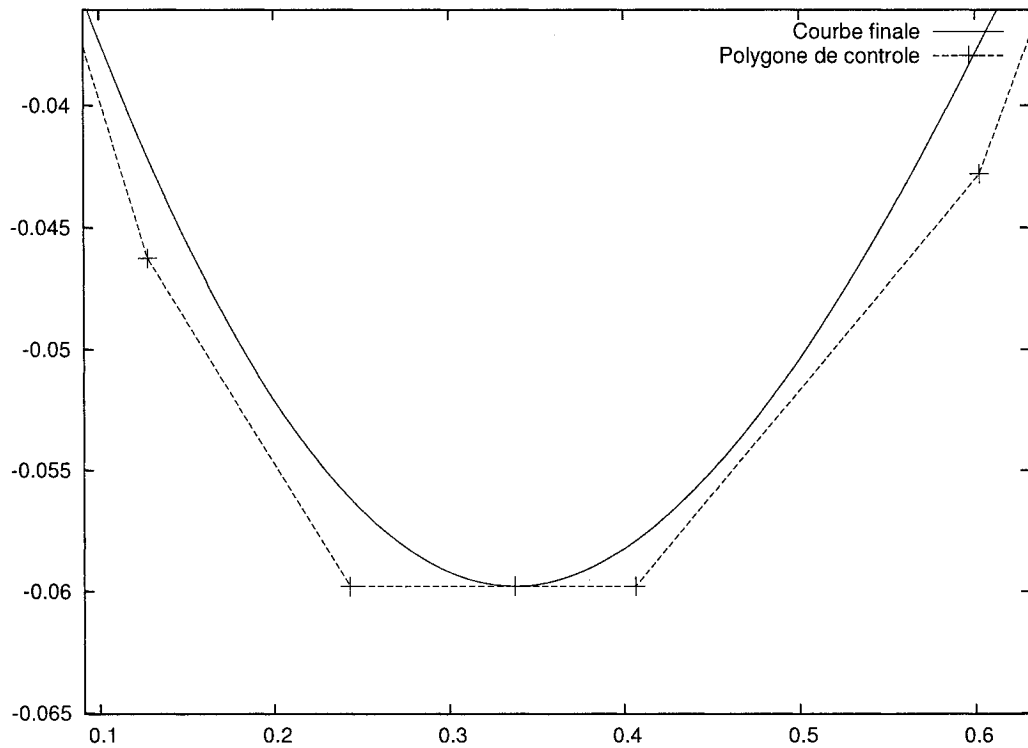


FIG. 3.5: Exemple d'un point de jonction issu du collage entre deux B-Splines primaires et distribution des points de contrôle autour de celui-ci. Remarquons que la distances séparant les point de contrôle colinéaires n'est pas la même à gauche et à droite.

Nous devons adapter la méthode décrite ci-haut pour permettre la construction d'une courbe avec le minimum d'informations disponibles. Pour ce faire, nous allons développer une méthode qui génère une B-Spline primaire avec seulement cinq points de contrôle. Le schéma est le même qu'à la figure 3.1, sauf qu'une des deux courbures est enlevée. Les conditions sur l'extrémité où la courbure est libre se limitent à la position et à l'angle de la pente. Les méthodes décrites plus haut pour générer les B-Splines primaires demeurent valides, mais quelques modifications doivent être apportées en ce qui concerne le vecteur nodal, comme il n'y a plus qu'un nœud à déterminer. Généralement, il suffit d'enlever le terme correspondant dans la sommation qui détermine la position du nœud (équations 3.16 et 3.19).

Afin de nous assurer que la courbe résultante soit continue en courbure, les deux courbes primaires de toutes jonctions doivent avoir la même valeur de courbure. Dans le cas où celle-ci est spécifiée, il n'y a pas de problème. Mais lorsque que la courbure est laissée libre, il faut se doter d'un moyen d'en déterminer une qui donnera à la courbe la continuité voulue et une belle allure (courbure imposée dans le même ordre que les courbures du voisinage). Pour ce faire, nous générons d'abord une B-Spline primaire de construction de 6 points de contrôle entre deux points clés qui ont leur courbure spécifiée. À partir de celle-ci, aux mêmes valeurs d'avancements u que le ratio des longueurs entre les points clés sautés, nous extrayons les valeurs de courbure locale. Ces courbures sont alors imposées aux points de jonction, donc aux points clés, ce qui assure la continuité et une belle allure à la courbe résultante.

Résumé

En résumé, nous savons maintenant comment procéder pour générer une B-Spline passant par plusieurs points clés. À partir de leurs contraintes géométriques (position & pente et, en option, courbure & amplitude), les dérivées et les nœuds internes du vecteur nodal sont déduits. De ces dernières, les point de contrôles de chaque B-Splines de base sont positionnés. Enfin, un collage faisant intervenir une normalisation donne la courbe finale.

3.1.2 B-Spline multi-contraintes en plusieurs endroits

Dans un deuxième temps, une approche globale est proposée afin d'obtenir une courbe passant par plusieurs points clés. Cette méthode comporte à peu près les mêmes étapes que la précédente, à savoir, déterminer un vecteur nodal et les dérivées à l'aide de l'information géométriques et de positionner les points de contrôle.

Cette méthode agit globalement. Elle a donc l'avantage de tout déterminer en une seule

grande étape, ce qui permet de diminuer le nombre de points de contrôle de la courbe résultante. En effet, ici nous avons autant de points de contrôle que de contraintes. Alors que dans la méthode précédente, l'opération de jonction a pour conséquence des nœuds triplés et résulte donc en davantage de points de contrôle. La conséquence des nœuds triples est le stockage de plus d'information que nécessaire pour définir la courbe, un peu comme dans le cas où l'utilisation de plusieurs courbes de Bézier est privilégiée au détriment d'une B-Spline (voir les sections 1.1.6 et 1.1.7). Comme inconvénient cependant, le prétraitement (changer les points clés en contraintes utilisable par le système), en plus de contenir une étape supplémentaire, est plus fastidieux. De plus, dès que nous voulons optimiser la courbure pour qu'elle soit le plus lisse possible, les temps de calcul s'allongent.

Cette approche est inspirée d'une méthode de manipulation de courbes par ajout de contraintes (*Constraint-based curve modification*), tel que nous le propose Piegl et Tiller (1995) (la théorie à ce sujet se trouve à l'annexe I), ainsi que du problème d'interpolation par une B-Spline. La méthode citée à l'annexe I.4 a été développée pour un cadre de manipulations interactives. Dans ce cas, les modifications sont donc apportées par un utilisateur sur une courbe existante. Généralement, le designer choisit à l'écran le ou les points sur lesquels il veut imposer des contraintes. L'outil calcule automatiquement dans un premier temps la valeur des paramètres u_r correspondant aux points choisis par des algorithmes d'inversions. Ensuite, les caractéristiques de la courbe $\mathbf{D}^{(k)}(u_r)$ sont extraites à ces points et comparées aux contraintes pour déterminer les différences $\Delta \mathbf{D}^{(k)}(u_r)$ à imposer, qui constituent le membre de droite du système à résoudre. Une fois solutionné, les déplacements des points de contrôle touchés sont effectués pour obtenir la courbe répondant aux contraintes.

La grande différence entre le cas cité à l'annexe I et la situation présente, est que nous n'avons pas de courbe originale à modifier. Nous utilisons donc une courbe initiale dont tous les points de contrôle sont situés à l'origine, donc toutes les dérivées sont nulles.

Sous cet angle, les différences à imposer sont alors directement les valeurs de contraintes, car $\mathbf{D}^{(k)}(u_r) = 0$ pour tous k et u_r . Aussi, le déplacement des points de contrôle devient leurs positions, car $\mathbf{P}_i = \emptyset$ quelque soit i . La notation $\mathbf{D}^{(k)}(u_r) + \Delta\mathbf{D}^{(k)}(u_r) = \sum_{i=0}^n N_{i,p}^{(k)}(u_r)(\mathbf{P}_i + \Delta\mathbf{P}_i)$ (voir équation I.15) se réduit donc à :

$$\Delta\mathbf{D}^{(k)}(u_r) = \sum_{i=0}^n N_{i,p}^{(k)}(u_r)\Delta\mathbf{P}_i \quad (3.24)$$

Où, en écriture compacte :

$$\mathbf{D} = [\mathbf{B}]\mathbf{P} \quad (3.25)$$

Nous construisons un système de $(n+1) \times (n+1)$, où il y a autant d'inconnus que d'équations. Le système est donc pleinement déterminé. La solution s'obtient par inversion de la matrice $[\mathbf{B}]$:

$$\mathbf{P} = [\mathbf{B}]^{-1}\mathbf{D} \quad (3.26)$$

Il reste cependant à construire un bon vecteur nodal U répondant au besoin de la courbe et à associer chaque point clé à un paramètre d'avancement \bar{u}_r . Ces derniers peuvent être des inconnus dans le système. Cependant, en procédant de cette façon, en plus d'augmenter le nombre total d'inconnues, nous obtenons un système fortement non linéaire dans lequel il faut itérer sur les \bar{u}_r . C'est pourquoi il est préférable d'avoir une méthode permettant de les fixer avant de procéder à l'imposition des contraintes. Le vecteur nodal doit aussi tenir compte des caractéristiques géométriques des points clés afin que la courbe résultante ait de douces variations de courbure, et donc une belle allure. C'est ce que nous présentons dans les prochains paragraphes.

La première étape consiste à comptabiliser le nombre total de contraintes N qui provient du nombre de points clés l et du nombre de contraintes par point clés f_i où $0 \leq i < l$. Le nombre total de contraintes N fixe la taille du système et donc le nombre de points

de contrôle. Il faut faire attention, car $n = N - 1$. Ensuite, on associe à chaque point clé, un paramètre d'avancement \bar{u}_r . Pour ce faire, nous utilisons la méthode centripète. Nous devons donc calculer la racine carrée de la distance entre chaque paire de points clés consécutifs et la sommer. Ceci donne :

$$L_{mod} = \sum_{i=0}^{n-1} \sqrt{|\mathbf{Q}_{i+1}^0 - \mathbf{Q}_i^0|} \quad (3.27)$$

et pour les \bar{u}_r :

$$\begin{aligned} \bar{u}_0 &= 0 \\ \bar{u}_{l-1} &= 1 \\ \bar{u}_i &= \bar{u}_{i-1} + \frac{\sqrt{|\mathbf{Q}_{i+1}^0 - \mathbf{Q}_i^0|}}{L_{mod}} \quad \text{Pour } 1 \leq i < l \end{aligned} \quad (3.28)$$

Ensuite, il faut générer le vecteur nodal, lequel sert à constituer la matrice $[\mathbf{B}]$ et le membre de droite \mathbf{D} . Toutes ces étapes sont présentées dans les sections suivantes.

3.1.2.1 Construction du vecteur nodal

En plus de comptabiliser la longueur des différents segments, nous conservons aussi les variations d'angle et de courbure à tous ces segments, lorsque l'information est disponible. Celle-ci va servir de base à la construction du vecteur nodal. Cette étape est la plus complexe du processus. Comme le degré p est fixé, tout comme le nombre de points de contrôle n , la taille du vecteur nodal m l'est aussi, et donc le nombre de nœuds internes. Les nœuds internes sont par définition ceux qui ne sont pas sur les bornes et sont généralement différents de 0 et de 1. Leurs nombre, n_{ni} , est la quantité d'inconnues à déterminer pour la construction du vecteur nodal et il est égal à $N - 4$. En effet, pour $p = 3$, il y a $2(p + 1) = 8$ nœuds sur les bornes. Il en reste donc $m - 2(p + 1) + 1 = m - 7$ qui ne sont pas sur les bornes. Or, $m = n + p + 1$ et $n = N - 1$, en substituant ces relations dans

TAB. 3.1: Association du nombre de nœuds du vecteur nodal en fonction de l'ordre des points clés

Points clés		Q_0^2	Q_1^1	Q_2^0	Q_3^2	Q_4^1	
Ordre		2	1	0	2	1	
Nb nœuds associés	Borne 0	3	2	1	3	2	borne 1
Vecteur nodal	$u_0, u_1,$ 0, 0,	$\overbrace{u_2, u_3, u_4},$ 0, 0, $u_4,$	$\overbrace{u_5, u_6},$ $u_5, u_6,$	$\overbrace{u_7},$ $u_7,$	$\overbrace{u_8, u_9, u_{10}},$ $u_8, u_9, u_{10},$	$\overbrace{u_{11}, u_{12}},$ 1, 1,	u_{13}, u_{14} 1, 1
Nb nœuds à déterminer		1	2	1	3	0	

la précédente, nous obtenons $n_{ni} = N - 1 + p + 1 - 2(p + 1) + 1 = N - 4$, ce que nous voulions montrer.

À chaque point clé, nous allons associer autant de nœuds dans le vecteur nodal que l'ordre additionné de 1. Ainsi, un point clé d'ordre $f = 2$ générera 3 nœuds dans le vecteur nodal de la courbe. Le lecteur attentif se rendra compte que le nombre de nœuds associés ne correspond pas avec le nombre de nœuds internes. Les deux points clés extrêmes seront traités différemment afin que ces nombres deviennent cohérents. Nous allons voir la façon dont sont associés les nœuds aux points clés par le biais d'un exemple. Considérons cinq points clés, $l = 5$, de différents ordre chacun. Dans l'exemple, il y a onze contraintes en tout ($N = 11$ et $n = 10$) donc sept nœuds internes à déterminer sur un vecteur nodal comptant quinze nœuds en tout (u_0 à u_{14}).

Le tableau 3.1 nous éclaire quant à l'association des nœuds aux points clés. Il reste maintenant à élaborer un moyen afin de déterminer la valeur des nœuds en fonction des contraintes à leur point clé associé pour assurer un nombre de points de contrôle suffisant au voisinage de chaque Q_i^f . En effet, il doit toujours y avoir assez de points de contrôle actifs autour d'un point clé pour assurer ses contraintes et il ne doit pas en générer trop pour éviter d'avoir des points de contrôle libres ou l'apparition d'oscillations. Par point de contrôle actif, nous parlons de ceux dont la fonction de base n'est pas nulle

à l'avancement \bar{u}_r associé au point clé. Nous appelons ce requis corrélation point clé-contrôle. Nous savons qu'à chaque intervalle paramétrique, il y a au plus 4 points de contrôle actifs. Alors, pour un point clé d'ordre $f = 2$, nous voudrions que ces 4 points de contrôle soient dans son voisinage.

La façon de déterminer les nœuds s'appuie à la fois sur le paramètre d'avancement des point clés, sur leur ordre et sur les variations d'angle et de courbure entre chacun d'entre eux. Dans le cas où l'ordre $f = 0$, il n'y a qu'un nœud à déterminer. La valeur de celui-ci est directement imposée à la valeur du paramètre d'avancement \bar{u}_r de son point clé associé. Alors, $u_j = \bar{u}_i$ où u_j est le $j^{\text{ième}}$ nœud interne à trouver du vecteur nodal et \bar{u}_i est le paramètre d'avancement associé au $i^{\text{ième}}$ point clé. Notons que si tous les points clé ont un ordre de 0, le vecteur nodal sera identique à celui construit pour l'interpolation par une B-Spline en utilisant la méthode centripète.

Dans le cas où l'ordre $f = 1$, deux nœuds sont à déterminer, deux conditions sont donc nécessaires. La première provient de la corrélation point clé-contrôle. Nous imposons que la moyenne de ces deux nœuds soit égale au paramètre d'avancement \bar{u}_r associé. La seconde provient de la variation d'angle au point clé. Plus la variation d'angle est grande, plus les nœuds doivent être près de \bar{u}_r et si elle est nulle, les nœuds seront à une distance maximale l'un de l'autre. Cette distance maximale est la demi distance moyenne entre l'avancement \bar{u}_r courant et les valeurs des voisins \bar{u}_{r+1} et \bar{u}_{r-1} . Le modificateur de la distance maximale qui répond aux exigences est l'opposée de la fonction exponentielle, e^{-x} , tel que représentée à la figure 3.6. Cette fonction à l'avantage de valoir l'unité si l'argument est nul et décroît au fur et à mesure que l'argument augmente. En équation nous obtenons, en définissant d'abord la direction γ :

$$\gamma = \arctan \left(\frac{Q_{i+1}^{0y} - Q_i^{0y}}{Q_{i+1}^{0x} - Q_i^{0x}} \right) \quad (3.29)$$

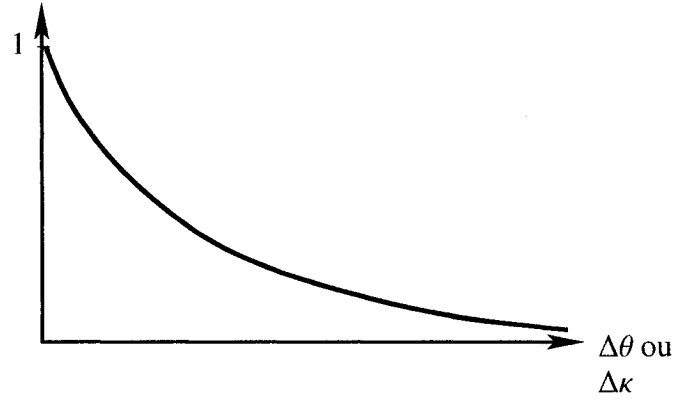


FIG. 3.6: Fonction e^{-x} servant à définir la distance entre deux nœuds du vecteur nodal

Puis les valeurs temporaires δ :

$$\delta_0 = |\mathbf{Q}_i^{1\epsilon} - \gamma| \quad \text{et} \quad \delta_1 = |\mathbf{Q}_{i+1}^{1\epsilon} - \gamma| \quad (3.30)$$

Et enfin la variation de la pente $\Delta\theta$:

$$\Delta\theta_i = \delta_0 + \delta_1 \quad (3.31)$$

Nous pouvons maintenant exprimer les distances :

$$\begin{aligned} d_{max} &= \frac{\frac{\bar{u}_{i+1} - \bar{u}_i}{2} + \frac{\bar{u}_i - \bar{u}_{i-1}}{2}}{2} = \frac{\bar{u}_{i+1} - \bar{u}_{i-1}}{4} \\ d &= d_{max} \cdot e^{(-\Delta\theta_i)} \end{aligned} \quad (3.32)$$

Et finalement, les deux nœuds u_j et u_{j+1} sont reliés par les relations

$$\begin{aligned} \bar{u}_i &= \frac{u_j + u_{j+1}}{2} \\ d &= u_{j+1} - u_j \end{aligned} \quad (3.33)$$

Dans le cas où l'ordre $f = 2$, il y a trois nœuds à déterminer avec trois conditions. Ici en-

core la corrélation point clé-contrôle et la variation de pente détermine deux conditions. La troisième provient quant à elle de la variation de courbure. La variation d'angle est calculée comme dans le cas précédent, mais nous calculons aussi la variation de la courbure, qui doit avoir le même effet sur la distribution des nœuds que la variation de pente. Ici aussi plus la variation est grande, plus nous voulons que les nœuds soient rapprochés et nous voulons qu'ils soient situés à une distance maximale lorsque la variation tombe à zéro. Nous calculons la variation de courbure de la façon suivante, avec les valeurs temporaires χ :

$$\chi_0 = \begin{cases} \kappa_i & \text{si } f = 2 \\ 0 & \text{Autrement} \end{cases} \quad \text{et} \quad \chi_1 = \begin{cases} \kappa_{i+1} & \text{si } f = 2 \\ 0 & \text{Autrement} \end{cases} \quad (3.34)$$

Et la variation de courbure $\Delta\kappa$:

$$\Delta\kappa_i = |\chi_0 - \chi_1| \quad (3.35)$$

Cette fois, au lieu de moyenner les distances, nous les utilisons séparément, donc :

$$\begin{aligned} d_{1_{max}} &= \frac{\bar{u}_i - \bar{u}_{i-1}}{2} \\ d_{2_{max}} &= \frac{\bar{u}_{i+1} - \bar{u}_i}{2} \\ d_1 &= d_{1_{max}} \cdot \left(\frac{e^{(-\Delta\theta_{i-1})} + e^{(-\Delta\kappa_{i-1})}}{2} \right) \\ d_2 &= d_{2_{max}} \cdot \left(\frac{e^{(-\Delta\theta_i)} + e^{(-\Delta\kappa_i)}}{2} \right) \end{aligned} \quad (3.36)$$

Enfin, les relations unissant les trois nœuds à déterminer sont :

$$\begin{aligned} \bar{u}_i &= \frac{u_j + u_{j+1} + u_{j+2}}{3} \\ d_1 &= u_{j+1} - u_j \\ d_2 &= u_{j+2} - u_{j+1} \end{aligned} \quad (3.37)$$

Finalement, pour les points de contrôle extrêmes, ils ne génèrent des nœuds que dans

le cas où le point clé a un ordre de $f = 2$. Si cette situation est rencontrée, le nœud est généré de façon similaire à ce que nous avons vu plus haut. Les distances sont exprimées par :

$$\begin{aligned} d_{max} &= \bar{u}_1 \quad \text{ou} \quad d_{max} = 1 - \bar{u}_{l-2} \\ d &= d_{max} \cdot e^{-\Delta\theta_i} \end{aligned} \quad (3.38)$$

Et la valeur du nœud :

$$u_j = d \quad \text{ou} \quad u_j = 1 - d \quad (3.39)$$

Selon que l'on se trouve au début ou à la fin de la courbe et $\Delta\theta_i$ se calcule comme aux équations 3.29 à 3.31.

En imposant un nombre de nœuds en fonction de l'ordre du point clé et leurs valeurs en fonction des caractéristiques géométriques, nous nous assurons que le système soit cohérent au moment de sa construction. En effet, chaque point clé se situe dans des intervalles distincts, lesquels sont suffisamment éloignés les uns des autres afin d'assurer assez de liberté à chaque point clé. De plus, dès que l'ordre $f \geq 1$, les nœuds du vecteur nodal sont différents des paramètres d'avancement \bar{u}_r . Ceci a pour conséquence que le maximum de points de contrôles sont actifs pour contraindre les caractéristiques d'un point clé, lequel a une plus grande influence, ce qui diminue les oscillations sur la courbe résultante.

3.1.2.2 Construction du membre de droite

La construction du membre de droite du système se résume à faire la transposition des contraintes en valeur de dérivées, comme aux équations 3.11 à 3.13 de la section 3.1.1.1. Seulement ici, ce sont tous les ordres de toutes les dérivées à imposer sur l'ensemble des points clés qui sont calculés. Nous devons donc les trier et conserver cet ordre tout au long du calcul. Les contraintes sont d'abord listées selon l'index i du point clé associé

et ensuite selon l'ordre f . Ainsi, pour l'exemple précédent, le membre de droite aura l'allure suivante :

$$\begin{aligned} RHS_X &= [x_0 \ x'_0 \ x''_0 \ x_1 \ x'_1 \ x_2 \ x_3 \ x'_3 \ x''_3 \ x_4 \ x'_4]^T \text{ et} \\ RHS_Y &= [y_0 \ y'_0 \ y''_0 \ y_1 \ y'_1 \ y_2 \ y_3 \ y'_3 \ y''_3 \ y_4 \ y'_4]^T \end{aligned} \quad (3.40)$$

Dans cette méthode, comme la courbe est construite en une seule étape, les amplitudes adimensionnelles sont multipliées par une longueur de référence, en opposition de la longueur de la courbe. Cette longueur de référence est la corde du profil associé aux points clés à spliner. L'avantage de procéder ainsi est de ne pas avoir à itérer sur la longueur de la courbe, ce qui accélère les calculs, mais qui donne aux valeurs d'amplitude une plus grande influence sur l'allure de la courbe finale.

3.1.2.3 Remplissage de la matrice du système

Nous devons maintenant assembler la matrice de dimension N , à partir de laquelle seront extraites les positions des points de contrôle. À l'aide du vecteur nodal de la courbe, les fonctions de bases sont d'abord générées et nous évaluons ces dernières ainsi que leurs dérivées d'ordre un et deux à tous les paramètres d'avancement \bar{u}_r . Or, nous savons qu'un maximum de $p + 1 = 4$ fonctions de bases sont non nulles pour un intervalle donné. Il en résulte que la matrice aura l'apparence d'une matrice diagonale. Dans notre exemple, elle est donnée par $[B] =$

$$\begin{bmatrix}
 N_{0,3}(0) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 N'_{0,3}(0) & N'_{1,3}(0) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 N''_{0,3}(0) & N''_{1,3}(0) & N''_{2,3}(0) & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & N_{2,3}(\bar{u}_1) & N_{3,3}(\bar{u}_1) & N_{4,3}(\bar{u}_1) & N_{5,3}(\bar{u}_1) & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & N'_{2,3}(\bar{u}_1) & N'_{3,3}(\bar{u}_1) & N'_{4,3}(\bar{u}_1) & N'_{5,3}(\bar{u}_1) & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & N_{4,3}(\bar{u}_2) & N_{5,3}(\bar{u}_2) & N_{6,3}(\bar{u}_2) & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & N_{6,3}(\bar{u}_3) & N_{7,3}(\bar{u}_3) & N_{8,3}(\bar{u}_3) & N_{9,3}(\bar{u}_3) & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & N'_{6,3}(\bar{u}_3) & N'_{7,3}(\bar{u}_3) & N'_{8,3}(\bar{u}_3) & N'_{9,3}(\bar{u}_3) & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & N''_{6,3}(\bar{u}_3) & N''_{7,3}(\bar{u}_3) & N''_{8,3}(\bar{u}_3) & N''_{9,3}(\bar{u}_3) & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & N_{10,3}(1) \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & N'_{9,3}(1) & N'_{10,3}(1)
 \end{bmatrix} \quad (3.41)$$

Il ne reste plus qu'à factoriser la matrice et multiplier le résultat par les membres de droite (RHS_X et RHS_Y) pour obtenir les deux composantes des points de contrôle, $\mathbf{P} = [\mathbf{B}]^{-1} \mathbf{C}^{(k)}$. Nous notons que sur cette matrice, chaque colonne contient le vecteur des fonctions de base ou une de ses dérivées évaluée à un paramètre \bar{u}_r et qu'elle est associée à un point de contrôle. Nous voyons ainsi que, toujours pour notre exemple, le premier point de contrôle (première colonne) intervient pour les trois contraintes qui y sont associées (position, pente, courbure). Aussi, certaines colonnes sont sollicitées pour deux \bar{u}_r différents. Ceci indique que les points de contrôle associés à ces colonnes sont actifs dans les intervalles d'influence de deux points clés.

Résumé

En résumé, nous disposons d'une deuxième méthode permettant de générer une B-Spline passant par plusieurs points clés. À partir de leurs contraintes géométriques (position et en option pente, courbure & amplitude), les dérivées et les nœuds internes du vecteur nodal sont déduits. À l'aide ces dernières, un système de dimension égale aux nombres de contraintes permet de déduire la position des points de contrôles de la B-Spline voulue.

3.1.3 Optimisation des amplitudes des dérivées premières

Dans les deux méthodes étudiées (section 3.1.1 et 3.1.2), nous pouvons spécifier ou laisser libre les amplitudes de la dérivée première. Comme les amplitudes ont une grande influence sur l'allure de la courbe finale, il est aussi possible de les optimiser afin d'obtenir la courbe la plus régulière possible. Nous allons utiliser la méthode d'optimisation à gradient quasi-Newton BFGS pour cette tâche (II). La valeur de départ des amplitudes est l'unité, à moins que l'utilisateur n'en spécifie une différente.

Le critère d'optimisation est l'obtention d'une courbe la plus lisse possible, *i.e.* où les variations de courbure seront minimales. Pour y parvenir, nous construisons une fonction coût comprenant deux termes. Le premier somme les variations d'angles d'un point de contrôle à l'autre. Minimiser cette grandeur fait en sorte que le polygone de contrôle ne se recoupe pas. Le deuxième terme provient de la variation de courbure sur la courbe. Pour l'évaluer, nous subdivisons la courbe en h segments de longueur identique. À chaque point, la courbure est extraite et comparée à ses voisines. La norme de la différence de la courbure d'un point à l'autre est sommée puis adimensionnalisée afin que le résultats soit du même ordre de grandeur que le premier terme, *i.e.* $\sim 2\pi$ pour un profil complet. Minimiser cette grandeur répartit la courbure sur toute la longueur de la courbe et minimise donc les oscillations. Mis en équation, nous obtenons :

$$\begin{aligned} \min_{\mathbf{x}} f(\mathbf{x}) \\ \text{tel que : } 0.1 \leq x_i \leq 3.0 \end{aligned} \quad (3.42)$$

avec

$$\begin{aligned} f(\mathbf{x}) &= f_1(\mathbf{x}) + f_2(\mathbf{x}) \\ f_1(\mathbf{x}) &= \sum_{i=1}^l \Delta\theta_i \\ f_2(\mathbf{x}) &= \frac{1}{h^2} \sum_{i=0}^h \left| \frac{\kappa_{\hat{u}_i}}{L_{seg}} \right| \end{aligned} \quad (3.43)$$

où les $\Delta\theta_i$ sont calculés comme précédemment (équations 3.29 et 3.31), et les $\kappa_{\hat{u}_i}$ sont les courbures aux extrémités des h segments. La fonction coût programmée a un échantillonnage de $h = 1000$. Les temps d'optimisation des amplitudes varient de quelques secondes à quelques minutes en fonctions du nombre de point clés et de la quantité de contraintes sur chacun d'eux. Les bornes sont les mêmes pour les variables de design, $x^L = 0.1$ et $x^U = 3.0$. Ainsi limitées, les amplitudes demeurent dans une plage où leurs effets ne sont jamais négligeables ni trop importants.

3.1.4 Comparaison des méthodes

Nous allons tester ces deux méthodes, ainsi que la paramétrisation PARSEC sur un cas simple : la construction d'un profil avec cinq points clés. Les contraintes des ces points clés proviennent des cinq points extrêmes d'un profil adimensionnel RAE2822. Nous imposons treize contraintes géométriques, soit la position, l'angle de la pente et la courbure aux points minimum et maximum et sur le bord d'attaque et la position et la pente aux deux points du bord de fuite. Notons qu'aucune optimisation géométrique n'est faite dans cette section (voir le chapitre 5 à cet effet). Les paramètres désirés (position, pente, courbure) sont directement extraits de la discrétisation du profil. Ce cas n'est pris qu'à titre d'exemple afin d'illustrer les capacités des méthodes d'interpolation avec peu d'information géométrique.

Pour la paramétrisation PARSEC, nous voyons sur la figure 3.7 qu'avec seulement les caractéristiques des cinq points clés, le profil RAE2822 est bien approximé, *i.e.* qu'il capture l'essentiel de la forme. Les erreurs de représentation se situent surtout sur la partie arrière de l'intrados, près du point d'inflexion. Seulement sept contraintes — angles de la pente aux bord de fuite (2), position et courbure aux point minimum et maximum (4) et courbure au bord d'attaque (1) — sont imposées pour obtenir ce profil, le reste — pente au bord d'attaque et aux points minimum et maximum (3), position des bords

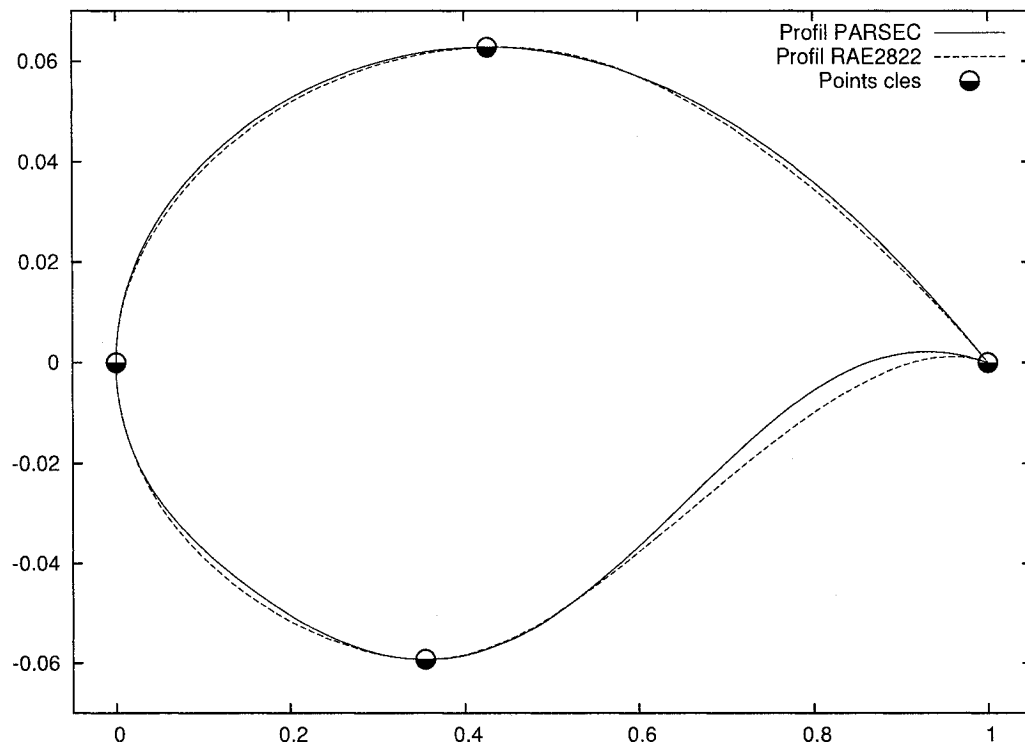


FIG. 3.7: Profil RAE2822 approximé à l'aide de la méthode PARSEC

d'attaque et de fuite (3) — étant obtenu par construction. Cependant, cette paramétrisation est victime de ses bons cotés, en effet il n'est possible que de représenter des profils adimensionnels. Plusieurs difficultés sont donc rencontrées lors de l'adaptation de cette paramétrisation pour un module de représentation de surfaces d'aile, ce que n'ont pas les méthodes basées sur les B-Splines. De plus, cette paramétrisation ne supporte que cinq points clés, caractérisant toujours les mêmes endroits (voir section 1.1.4). Il n'est pas possible de lui en ajouter de nouveaux, pas plus que de faire le transfert mathématique entre cette forme et une représentation par B-Spline, dû à l'emploi dans sa formulation d'exposants de grandes valeurs et qui ne sont pas des entiers.

La méthode de collage de B-Splines contraintes aux extrémités est représentée aux figures 3.8. Dans la figure 3.8(a), le profil est obtenu en laissant les amplitudes libres, donc leurs normes égale l'unité, et le profil de la figure 3.8(b) a ses amplitudes optimisées se-

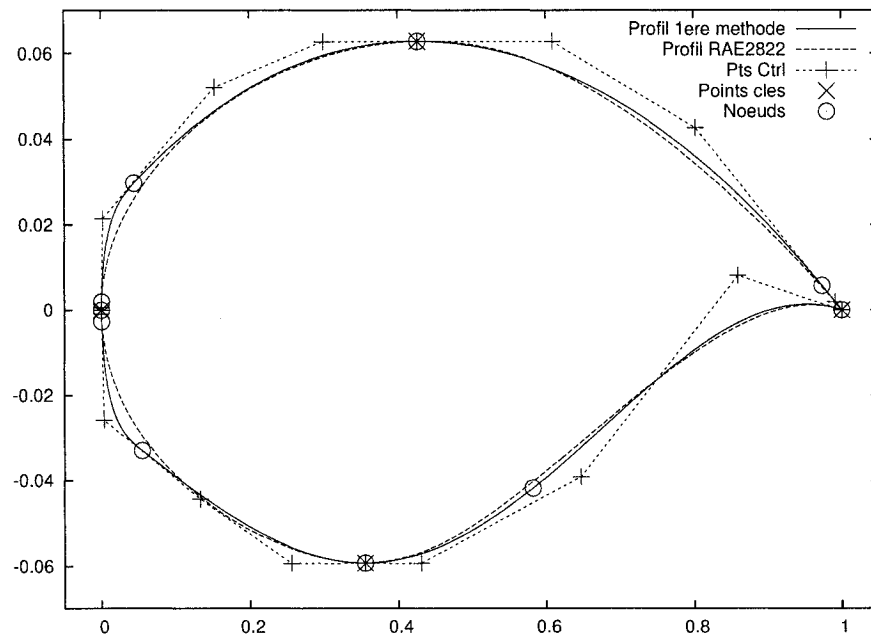
lon la méthode de la section 3.1.3. Les courbes B-Splines obtenues par cette technique comporte 19 points de contrôle dans ce cas, soit deux B-Splines de six points de contrôle (celles ayant leur jonction au bord d'attaque) et deux de cinq points de contrôle (celles passant par le bord de fuite, car la courbure n'est pas spécifiée en cet endroit) diminués de un point de contrôle par jonction, suite au collage. Le profil de la première figure semble bien approximer le profil RAE2822, les plus grandes erreurs de représentation se situent cette fois près du bord d'attaque. Avec l'optimisation automatique des amplitudes, nous améliorons la représentation vers le bord d'attaque, car la courbure devient plus régulière. Cependant, le lissage induit des erreurs de représentation plus grande vers la partie arrière de l'intrados, près du point d'inflexion, car la courbure y est mieux distribuée.

Enfin, la méthode imposant plusieurs contraintes géométriques est présentée aux figures 3.9. Ici encore, le profil de la première figure (3.9(a)) est obtenu en laissant les amplitudes à l'unité et le deuxième (3.9(b)) avec des amplitudes optimisées. Les B-Splines de la courbe résultante a treize points de contrôle, autant que de contraintes géométriques. Nous remarquons cependant sur la figure du haut que la représentation n'est pas très fidèle au profil RAE2822 entre les points clés, donnant des points de contrôle qui forme de multiples inflexions. L'optimisation automatique des amplitudes améliore grandement la situation (3.9(b)), mais la représentation demeurent quand même moins bonne que dans l'exemple précédent.

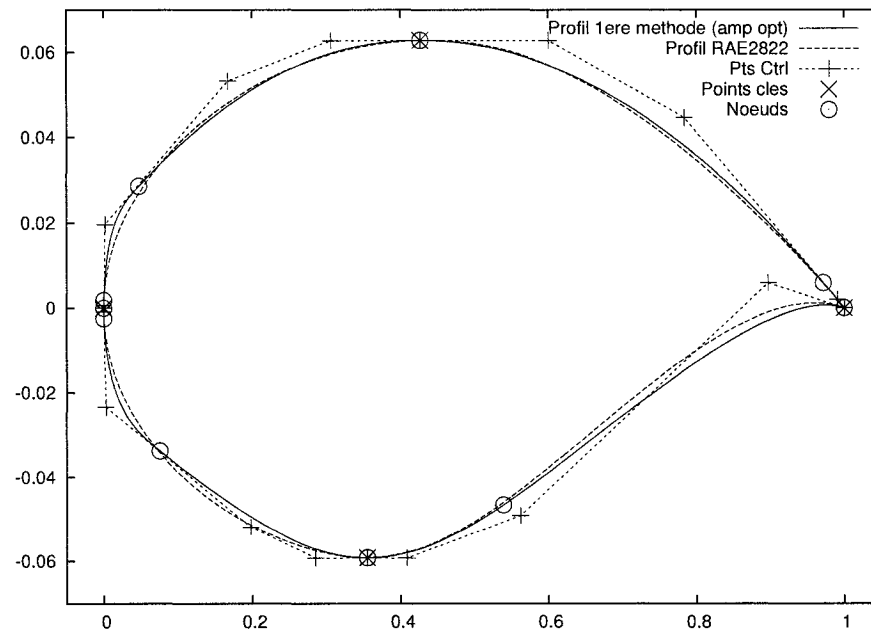
Résumé

En résumé, sans optimisation géométrique, la représentation PARSEC donne de bonnes approximations, mais ne peut être étendue, ce qui limite son utilisation. La méthode de collage de B-Splines contraintes aux extrémités semble donner de meilleurs résultats que celle imposant de multiples contraintes pour le cas de profils aérodynamiques. Pour la deuxième méthode, l'amélioration donnée par l'optimisation des amplitudes est grande, mais insuffisante. De plus, cette méthode s'est avérée moins robuste que celle par collage

de B-Splines. Pour les méthodes basées sur les B-Splines, comme les amplitudes sont d'autres variables de design, il est préférable que l'utilisateur (ou l'optimiseur géométrique) les prennent en charge au lieu qu'elles soient automatiquement choisies afin d'avoir un plus grand espace de design. Pour ces raisons, la méthode de collage de B-Splines contraintes aux extrémités sans optimisation des amplitudes est considérée comme la méthode à utiliser par défaut.

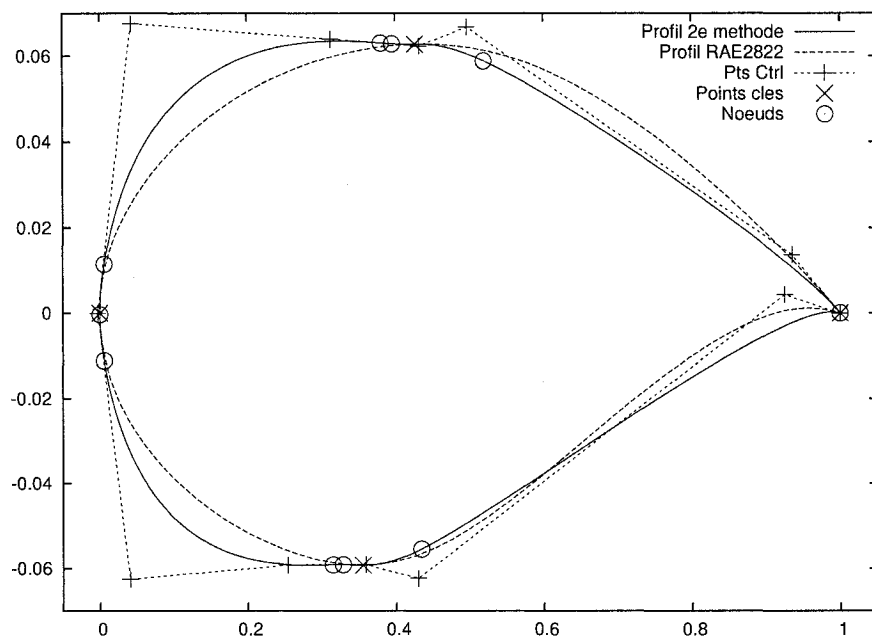


(a) Sans optimisation des amplitudes

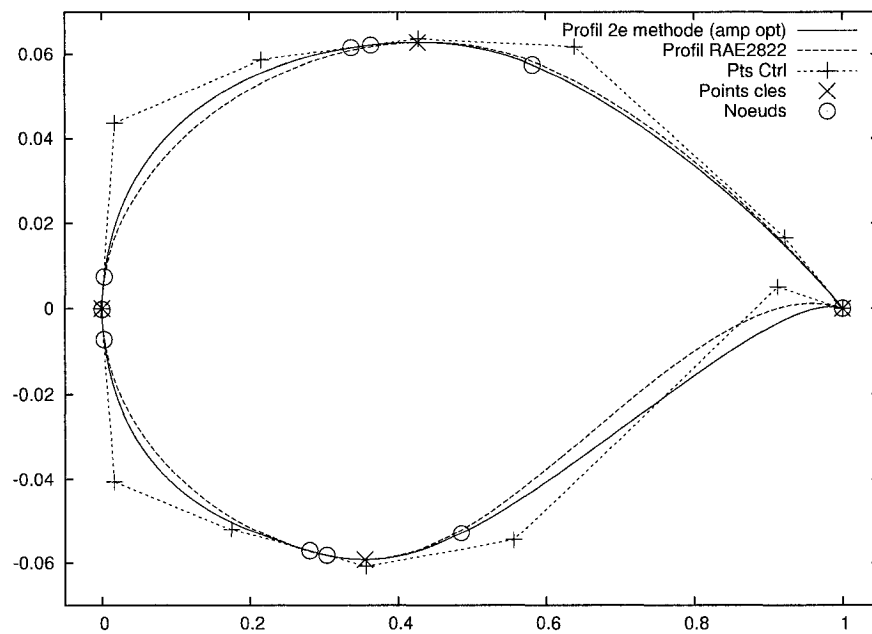


(b) Avec optimisation des amplitudes

FIG. 3.8: Première méthode : Profil RAE2822 approximé avec quatre B-Splines contraintes aux extrémités jointent ensemble



(a) Sans optimisation des amplitudes



(b) Avec optimisation des amplitudes

FIG. 3.9: Deuxième méthode : Profil RAE2822 approximé avec une B-Spline multi-contraintes en plusieurs endroits

3.2 Interpolation conique dans le sens de l'envergure

Une interpolation conique a été développée pour déterminer la variation des différents paramètres générateurs des profils dans le sens de l'envergure. Nous avons vu au chapitre 2 que ces paramètres ne sont connus qu'aux trois profils guides. Les coniques sont utilisées pour interpoler les paramètres à ces trois stations afin de pouvoir extraire leur valeur entre ceux-ci. L'utilisation des coniques a été retenue pour se conformer aux méthodes présentes chez BA.

Nous partons de la définition générale des coniques dans le but d'élaborer une méthode permettant de construire une courbe interpolant les paramètres donnés aux trois profils guides. Rappelons tout d'abord la forme générale des coniques :

$$ax^2 + bxy + cy^2 + dx + ey + f = 0 \quad (3.44)$$

Nous voyons dans l'équation 3.44 que six coefficients sont présents. Cependant, nous pouvons en faire disparaître un arbitrairement en divisant l'équation par celui-ci. Nous avons alors cinq coefficients à déterminer, cinq conditions sont donc nécessaires pour fermer le système. Comme nous avons déjà trois positions à interpoler, il nous faut trouver deux autres conditions pour complètement définir la conique. Dans le cas présent, ces deux conditions proviendront de la pente et de la courbure de la courbe d'interpolation au point situé à la cassure. Ainsi, la conique résultante va interpoler trois points, dont le point central avec des conditions relatives à la pente et la courbure, tel qu'illustré à la figure 3.10. Nous verrons dans les paragraphes suivants la façon de déterminer ces deux grandeurs. Pour différencier les points à interpoler, les indices utilisés pour les stations de l'emplanture, de la cassure et du bout sont respectivement e , c et b .

La pente imposée au point de la cassure est la même que celle de la droite reliant les

deux points limites (droite $D1$, voir figure 3.10). En imposant la pente de cette façon, nous nous assurons que la distance d entre la droite $D1$ et la courbe d'interpolation soit maximum au point de la cassure. Cette condition s'exprime de la façon suivante :

$$y'_c = \frac{y_b - y_e}{x_b - x_e} \quad (3.45)$$

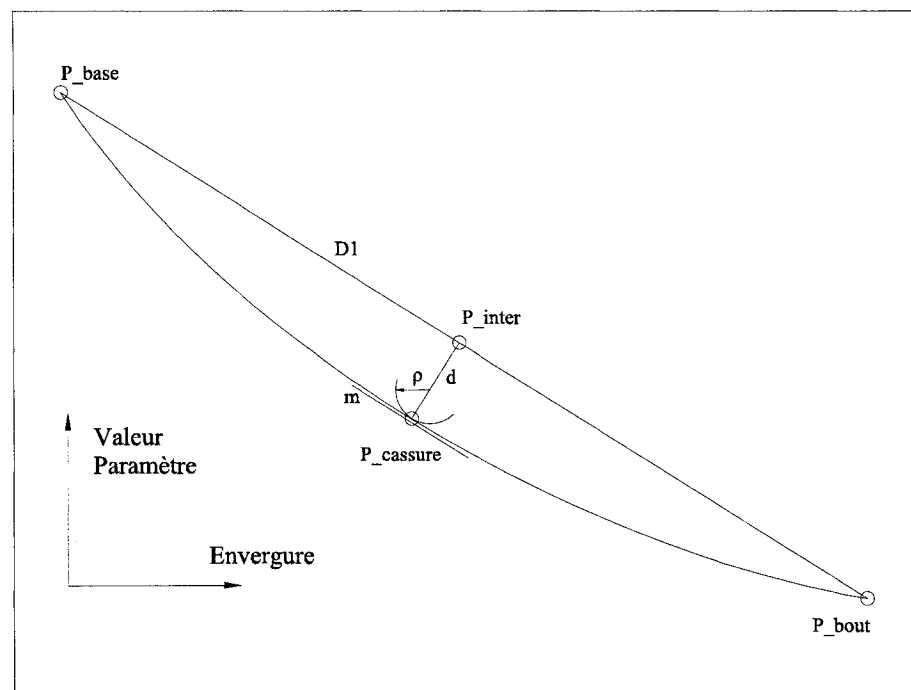


FIG. 3.10: Méthode de construction d'une conique interpolant trois points dont un avec pente et courbure

La grandeur de la courbure à imposer à ce point est plus difficile à déterminer. Il nous faut une condition qui, peu importe l'ordre de grandeur des valeurs des paramètres à interpoler, nous donne une courbe où les variations sont lisses et continues sur toute la longueur. Autrement dit, nous voulons éviter des cassures géométriques apparentes dues à de trop grandes variations très locale de la dérivée seconde. Une bonne façon de faire alors est d'imposer une courbure à la cassure de la même valeur que la variation d'angle

entre l'emplanture et le bout divisée par la longueur de la droite $D1$. Ainsi, plus le point à la cassure est près de la droite, moins les variations d'angles sont grandes et plus le rayon de courbure est grand, donc plus la conique se rapproche d'une droite. À l'opposé, si le point à la cassure est éloigné, la courbure va s'accroître. Mise en équation, avec l'équation 3.9, cette condition devient :

$$\rho = \frac{\Delta\theta}{L} = \frac{(1 + (y')^2)^{3/2}}{y''} \quad (3.46)$$

Alors,

$$y'' = \frac{(1 + (y')^2)^{3/2} L}{\Delta\theta} \quad (3.47)$$

avec

$$\begin{aligned} L &= \sqrt{(x_t - x_r)^2 + (y_t - y_r)^2} \\ \Delta\theta &= \pm |\theta_{deb} - \theta_{moy}| + |\theta_{moy} - \theta_{fin}| \end{aligned} \quad (3.48)$$

où

$$\begin{aligned} \theta_{deb} &= \arctan\left(\frac{y_c - y_e}{x_c - x_e}\right) \\ \theta_{fin} &= \arctan\left(\frac{y_b - y_c}{x_b - x_c}\right) \\ \theta_{moy} &= \arctan\left(\frac{y_b - y_e}{x_b - x_e}\right) \end{aligned} \quad (3.49)$$

Le signe de $\Delta\theta$ fixe le sens de la courbure et est déterminé par la position du point de la cassure par rapport à la droite $D1$. Une courbe à courbure positive tourne à gauche lorsque nous la parcourons, et tourne à droite pour une courbure négative.

Pour imposer ces deux nouvelles conditions, une expression des dérivées première et deuxième d'une conique générale est requise. Pour cela, partons de l'expression de la dérivée totale :

$$\frac{dF}{dx} = \frac{\partial F}{\partial x} + \frac{\partial F}{\partial y} \frac{dy}{dx} \quad (3.50)$$

Et posons F égal à l'expression 3.44.

Comme nous voulons une expression de la pente en tous points, nous isolons de l'équation 3.50 le terme dy/dx . Aussi, dans le présent cas, le terme de gauche vaut zéro par définition. En effet, comme la fonction est toujours égale à zéro, la dérivée totale de la conique va toujours être nulle aussi. Nous obtenons donc :

$$\frac{dy}{dx} = -\frac{\partial F/\partial x}{\partial F/\partial y} \quad (3.51)$$

Si nous continuons à pousser le même raisonnement pour obtenir la dérivée seconde, nous pouvons obtenir :

$$\begin{aligned} \frac{d^2F}{dx^2} &= \frac{d}{dx} \left(\frac{dF}{dx} \right) \\ &= \frac{d}{dx} \left(\frac{\partial F}{\partial x} + \frac{\partial F}{\partial y} \frac{dy}{dx} \right) \\ &= \frac{d^2y}{dx^2} \left(2 \frac{\partial F}{\partial y} \right) + \frac{dy}{dx} \left(2 \frac{\partial^2 F}{\partial x \partial y} + \frac{\partial^2 F}{\partial y^2} \frac{dy}{dx} \right) + \frac{\partial^2 F}{\partial x^2} \end{aligned} \quad (3.52)$$

Pour alléger les équations, utilisons la notation :

$$\begin{aligned} \frac{\partial F}{\partial y} &= F_y, & \frac{\partial F}{\partial x \partial y} &= F_{xy}, & \dots \\ \frac{dy}{dx} &= y', & \frac{d^2y}{dx^2} &= y'', & \dots \end{aligned} \quad (3.53)$$

Après réarrangement, nous obtenons une expression pour la dérivée seconde :

$$y'' = \frac{-F_{xx} - y'(2F_{xy} + F_{yy}y')}{2F_y} \quad (3.54)$$

Maintenant que nous avons les expressions des dérivées recherchées, nous pouvons élaborer le système d'équation permettant d'obtenir la conique voulue. À partir de l'équa-

tion 3.44, calculons les dérivées partielles d'une conique générale :

$$\begin{aligned}
 F_x &= 2ax + by + d \\
 F_y &= 2cy + bx + e \\
 F_{xx} &= 2a \\
 F_{yy} &= 2c \\
 F_{xy} &= b
 \end{aligned} \tag{3.55}$$

Nous avons maintenant toute l'information nécessaire pour construire un système permettant de déterminer la conique. Les cinq conditions comprennent les trois points à interpoler et la pente et la courbure sur un de ces points. Nous pouvons exprimer les cas généraux par ce système d'équations :

$$\begin{bmatrix}
 1 & 0 & 0 & 0 & 0 & 0 \\
 x_r^2 & x_r y_r & y_r^2 & x_r & y_r & 1 \\
 x_b^2 & x_b y_b & y_b^2 & x_b & y_b & 1 \\
 x_t^2 & x_t y_t & y_t^2 & x_t & y_t & 1 \\
 2x_b & x_b y'_b + y_b & 2y_b y'_b & 1 & y'_b & 0 \\
 2 & 2(x_b y''_b + y'_b) & 2(y_b'^2 + 2y_b y''_b) & 0 & 2y''_b & 0
 \end{bmatrix}
 \begin{bmatrix}
 a \\
 b \\
 c \\
 d \\
 e \\
 f
 \end{bmatrix}
 =
 \begin{bmatrix}
 1 \\
 0 \\
 0 \\
 0 \\
 0 \\
 0
 \end{bmatrix} \tag{3.56}$$

Notons que la première ligne ne contient que des 0 à l'exception de la première colonne, correspondant au coefficient de a , qui est fixé à $a = 1$ par défaut. Ce choix n'a pas d'importance, sauf quand la conique dégénère en droite. Quand cette situation se présente, les termes de second ordre sont nuls, et le choix de la variable à éliminer a des conséquences sur la conique. Nous voyons sur la figure 3.11 quelques exemples de coniques interpolant trois points avec la méthode décrite dans cette section.

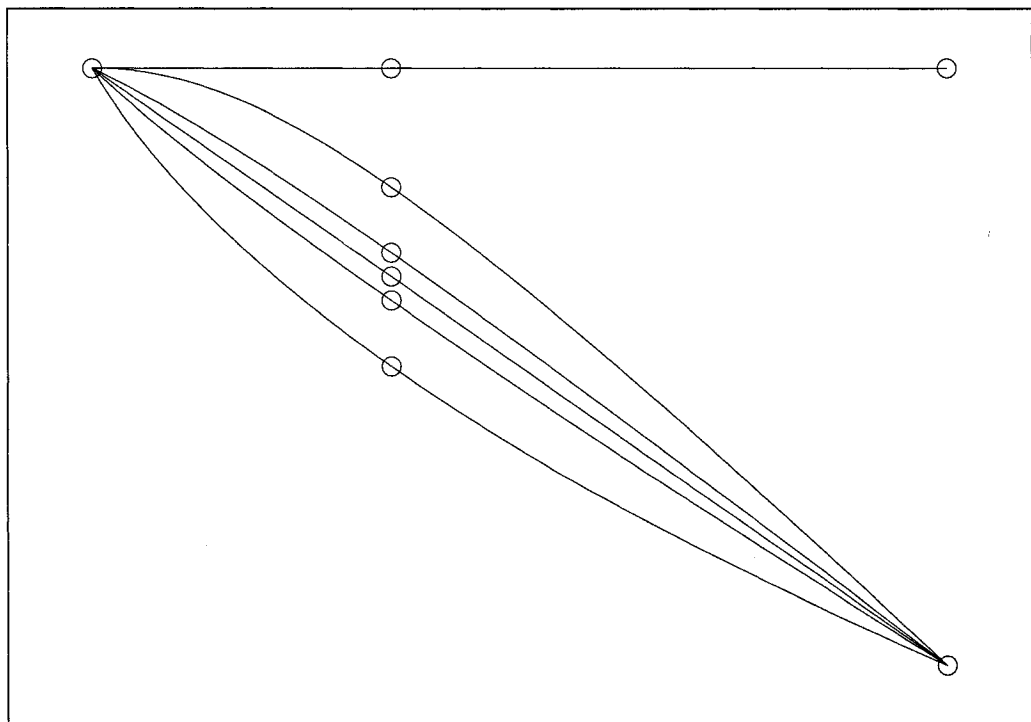


FIG. 3.11: Exemples de coniques interpolant trois points

CHAPITRE 4

CONSTRUCTION DE L'AILE

Tous les éléments nécessaires à la construction de l'aile sont rassemblés. Les objectifs, les contraintes et les différentes méthodes de construction des familles de courbes et de paramétrisation vont être appliqués afin de construire une aile transsonique actuelle. Ce chapitre contient une description des deux méthodes implantées au cours du projet. La première, basée sur la paramétrisation PARSEC, a permis de se familiariser avec les concepts de la construction de l'aile 3D. De plus, cette méthode nous a permis de mettre en évidence des difficultés que nous avons alors résolues dans l'implantation de la deuxième méthode. Cette dernière se base sur une définition hiérarchique des contraintes et sur l'utilisation avancée des B-Splines.

Les paramètres de construction sont traités par le programme qui construit un squelette 3D de caractéristiques géométriques devant être respecté. Par la suite, les courbes et surfaces sont appliquées sur ce squelette pour obtenir l'aile. Donc, en modifiant les paramètres, le squelette se déforme, entraînant par le fait même les courbes et surfaces à changer pour obtenir ainsi la nouvelle aile.

4.1 Premier module avec la définition PARSEC

Un premier module de génération de surface d'ailes d'avion a été conçu à l'aide de la paramétrisation PARSEC (voir section 1.1.4). L'objectif de cette implantation était de se familiariser avec les enjeux relatifs à la construction 3D d'une aile.

La méthode de construction nécessite la définition des paramètres PARSEC aux stations

à l'emplanture, à la cassure et au bout. Des interpolants sont utilisés pour extraire les valeurs des paramètres à toutes les autres stations. Le module comprend une interpolation splinaire et conique. Pour augmenter le contrôle des interpolants, les valeurs des pentes des paramètres aux extrémités sont aussi nécessaires.

La méthode utilise aussi la définition de la forme en plan. L'enveloppe extérieure de celle-ci est suffisante, car les contraintes intérieures ne peuvent être supportées par la définition PARSEC. De plus, l'angle de torsion aux trois stations de définition est nécessaire ainsi que des caractéristiques relatives à la discrétisation de l'aile finale (nombre de station, de point de discrétisation, paramètre de décalage, choix de l'interpolant, ...).

Dans un premier temps, le module lit les fichiers d'entrées et les charges en mémoire. Pour chacun des paramètres PARSEC spécifié aux trois stations de définition, un interpolant défini dans le fichier de construction est appliqué dans le sens de l'envergure. Cette opération est répétée pour tous les paramètres. En fonction du nombre de stations voulues dans l'aile, la position sur l'envergure des stations est extraite. Sur ces stations, sont interpolés tous les paramètres, pour y construire un profil PARSEC. À ce moment, la définition de la géométrie adimensionnelle 2D des profils de toutes les stations est en mémoire. Ensuite, à l'aide des facteurs d'échelle, de rotation et de translation, le module positionne chaque station dans l'espace pour constituer l'aile. C'est de cette dernière opération, toute simple en apparence, que proviennent quelques difficultés.

La rotation à appliquer au profil pour atteindre la station est extraite de la distribution de torsion. Le facteur d'échelle provient de la corde réelle, *i.e.*, de la corde apparente issue de la forme en plan divisée par le cosinus de l'angle d'incidence du profil, $c_r = c_x / \cos(\alpha)$. La translation est fixée selon la station et la position du bord d'attaque. Cependant, en procédant de la sorte, le bord d'attaque après la transformation n'est plus le point le plus en avant et n'est plus perpendiculaire à l'horizontale, comme le montre la figure 4.1. Bien que l'erreur commise ne soit pas très grande, la ligne du bord d'attaque

de l'aile n'est plus respectée, donc la forme en plan ne l'est plus également.

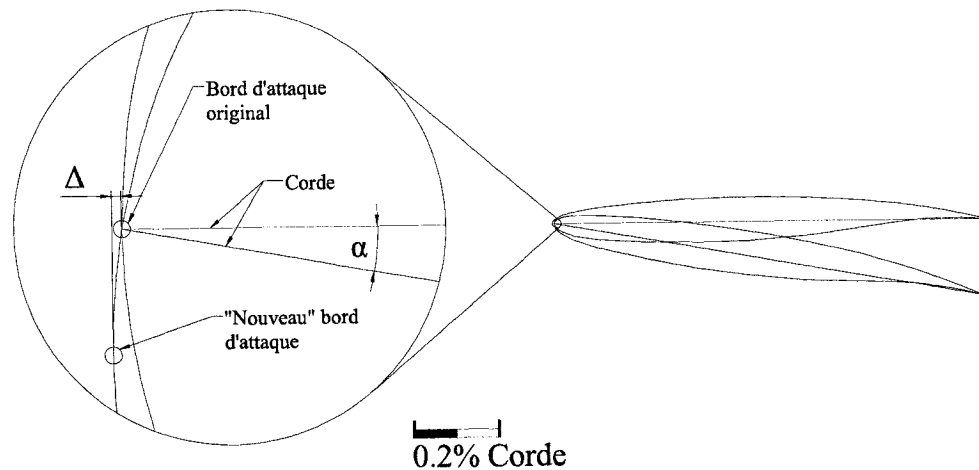


FIG. 4.1: Problèmes liés à la rotation d'un profil

Pour éviter ce problème, avec l'angle d'incidence comme information, nous pouvons trouver l'endroit sur le profil qui sera perpendiculaire à l'horizontale après la rotation. Ce point est alors défini comme le bord d'attaque et la corde est tirée entre ce point et le bord de fuite (voir figure 4.2). Celle-ci sert de nouvelle corde adimensionnelle. En appliquant les transformations solides à partir de ce nouveau bord d'attaque, nous nous assurons que la ligne du bord d'attaque est conservée droite et perpendiculaire à l'horizontale et que la forme en plan est respectée. C'est cette dernière méthode qui a été implémentée dans le module. Cependant, le rayon de courbure au bord d'attaque n'est pas conservé, celui que nous avons imposé se retrouvant près de la ligne du bord d'attaque, mais pas directement dessus.

Enfin, un dernier problème a été observé qui provient du fait que nous interpolons les paramètres adimensionnels, puisque la définition PARSEC est adimensionnelle. Notons, que la fonction représentant la longueur de la corde en fonction de l'envergure n'est pas

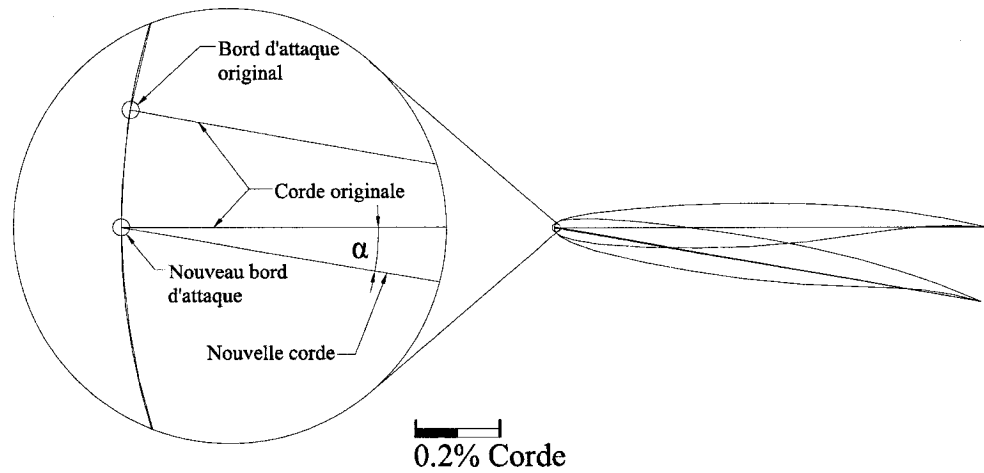


FIG. 4.2: Seconde approche pour la rotation d'un profil

continue (se référer à la figure 2.3). Comme cette longueur sert de base pour trouver le facteur d'échelle, une discontinuité se glisse à la cassure par construction, ce qui n'est pas très désirable.

Notons enfin que les interpolants implémentés se manipulent mal, car les pentes aux extrémités doivent être fournies. Or, ces valeurs n'ont pas un grand sens physique, et sont donc difficiles à évaluer et à manipuler. Comme nous voulons garder un sens physique aux paramètres de construction, ces derniers devront être mis de côté. La figure 4.3 illustre une discrétisation d'aile créée à l'aide de ce module.

4.2 Deuxième module utilisant des B-Splines

Pour éviter les problèmes du premier module et surtout pour construire une aile avec tous les requis et contraintes que nous avons définis au chapitre 2, une nouvelle approche est proposée. Celle-ci se base sur les B-Splines et utilise l'imposition de contraintes aux

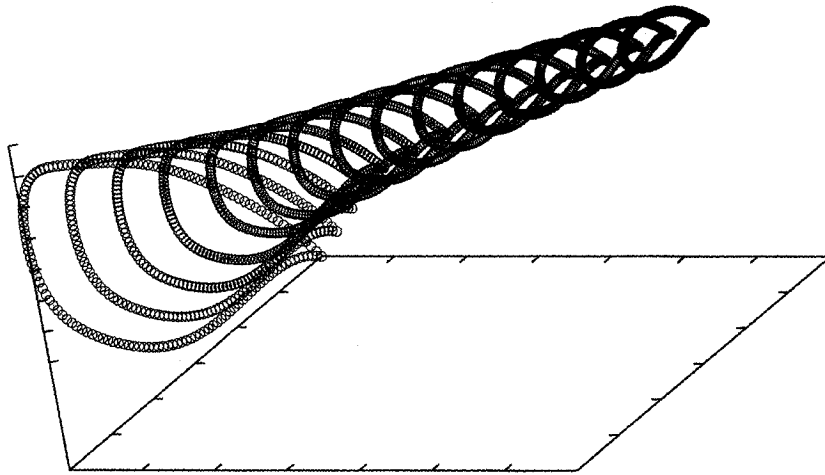


FIG. 4.3: Exemple d'aile construite avec le premier module

extrémités d'une courbe (section 3.1.1) comme méthode de construction du profil. De plus, l'interpolation dans le sens de l'envergure ne se fait qu'avec les coniques définies en 3.2. Le principal avantage provient du fait que seulement les trois valeurs de paramètres sont nécessaires, car les pentes ne sont plus nécessaires. En prenant cette définition, les valeurs des paramètres peuvent être extraites directement. Il est à noter que cette approche, tout comme la première, définit la surface extérieure de l'aile à l'aide d'un nombre variable de stations, et non pas par une représentation paramétrique explicite de la surface, comme dans les travaux de Bentamy (2003). Avant de voir en détail les principales étapes de construction de l'aile, nous présentons d'abord la façon dont les contraintes sont appliquées par couches pour définir un profil.

4.2.1 Contraintes multi-couches

En gardant en tête l'objectif de pouvoir faire des optimisations selon divers degrés de précision, les contraintes vont être appliquées par couche qu'il sera possible d'allumer ou d'éteindre selon les caractéristiques désirées, le degré de détail et le contrôle voulu. La stratégie consiste à employer une couche de base, toujours présente, sur laquelle d'autres couches peuvent être appliquées (voir la figure 4.4). En fonction des contraintes, le résultat est un profil adimensionnel ou une station avec un angle d'incidence quelconque et une corde non unitaire. Le profil est construit en partant du bord de fuite sur l'intrados, en parcourant celui-ci jusqu'au bord d'attaque, pour retourner au bord de fuite par l'extrados.

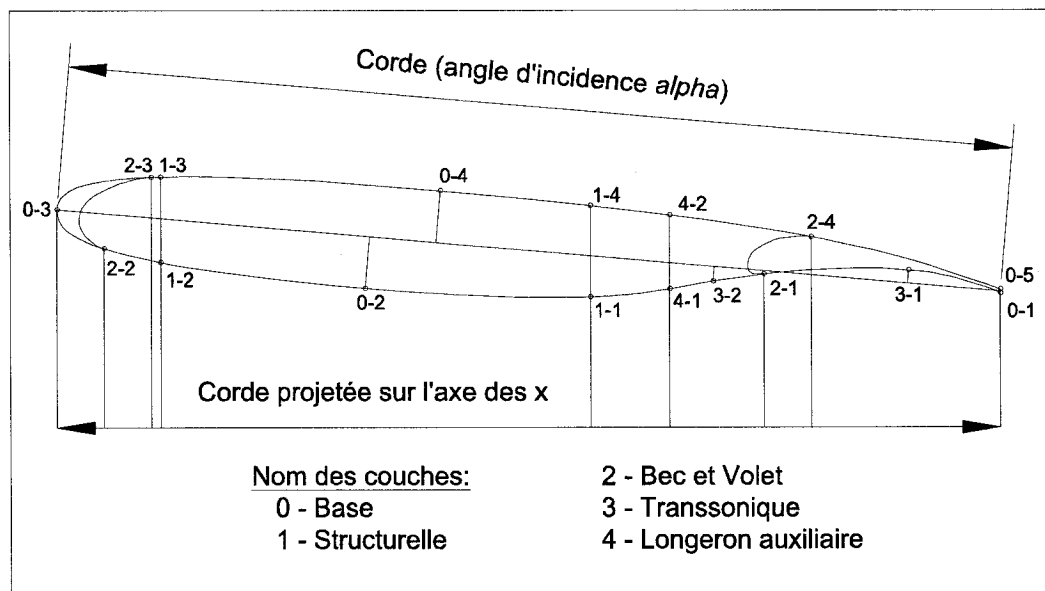


FIG. 4.4: Identification des différentes couches de paramètres pour contraindre une station d'aile

La couche de base, numéro 0, est constituée de cinq points clés et divise donc le profil en quatre parties. Les cinq points clés sont : le bord de fuite sur l'intrados (noté 0-1), le point minimum sur l'intrados (noté 0-2), le bord d'attaque (noté 0-3), le point maximum

sur l'extrados (noté 0-4) et le bord de fuite sur l'extrados (noté 0-5). Sur ces points, en plus de la position, la pente et la courbure sont nécessaires, sauf au bord de fuite où la courbure est optionnelle (voir le tableau 4.1). L'amplitude de la dérivée est optionnelle également. Il s'agit des mêmes caractéristiques que celles utilisées pour l'exemple de la section 3.1.4.

La couche numéro 1 contraint les positions des longerons. Elle compte quatre points clés, numérotés de 1-1 à 1-4. La pente à ces points est requise alors que la courbure et l'amplitude sont optionnelles. Sur la figure 4.4, la projection sur l'axe des x de ces points se confondent deux à deux, car nous voulons que les longerons soient perpendiculaires à l'écoulement en régime de croisière. Cette nécessité provient de considérations structurales afin de maximiser le ratio épaisseur/poids de l'aile.

La deuxième couche fixe les points où les bec et volet se confondent avec le profil de croisière. Elle compte quatre points clés, numérotés de 2-1 à 2-4. Ici encore, la position et la pente sont requises, alors que la courbure et l'amplitude sont optionnelles.

La couche numéro trois concerne les caractéristiques transsoniques, *i.e.*, le point correspondant à l'endroit le plus haut de la cavité et le point d'inflexion sur l'intrados. Pour la cavité (noté 3-1), seule la position est requise, la pente étant toujours alignée selon la corde du profil. Les autres caractéristiques sont optionnelles. Quant au point d'inflexion (noté 3-2), la position et la pente doivent être fournies, la courbure est par définition nulle et l'amplitude est optionnelle.

La dernière couche, numéro 4, ne concerne que la partie intérieure (*in-board*) de l'aile, *i.e.* entre l'emplanture et la cassure, là où un troisième longeron peut être présent. Les caractéristiques des ces deux nouveaux points, notés 4-1 et 4-2, sont les mêmes que ceux de la première couche.

TAB. 4.1: Tableau résumé des couches de contraintes

Couche	Numéro	Nom	Position	Pente	Courbure	Amplitude
0	1	TE _{IN}	Ø/ Req.	Req.	Opt.	Opt.
	2	IN	Req.	Ø	Req.	Opt.
	3	LE	Ø/ Req.	Ø	Req.	Opt.
	4	EX	Req.	Ø	Req.	Opt.
	5	TE _{EX}	Ø/ Req.	Req.	Opt.	Opt.
1	1	RSI	Req.	Req.	Opt.	Opt.
	2	FSI	Req.	Req.	Opt.	Opt.
	3	FSE	Req.	Req.	Opt.	Opt.
	4	RSE	Req.	Req.	Opt.	Opt.
2	1	FI	Req.	Req.	Opt.	Opt.
	2	SI	Req.	Req.	Opt.	Opt.
	3	SE	Req.	Req.	Opt.	Opt.
	4	FE	Req.	Req.	Opt.	Opt.
3	1	Cavité	Req.	Ø	Opt.	Opt.
	2	Inflexion	Req.	Req.	Ø	Opt.
4	1	SSI	Req.	Req.	Opt.	Opt.
	2	SSE	Req.	Req.	Opt.	Opt.

Ø : champs remplis par construction

Req. : champs requis

Opt. : champs optionnels

Profil 2D / Station

4.2.2 Construction des stations guides

La première étape de la construction de l'aile est la génération des trois profils aux stations de définitions. Comme les paramètres sont toujours donnés dans le fichier d'entrée de façon adimensionnelle pour un maximum de compréhension, ceux-ci sont dimensionnés par rapport à la corde réelle avant l'interpolation dans le sens de l'envergure. Cette approche permet d'éviter l'étape de dimensionnement des stations qui introduit une discontinuité et les problèmes induits par les transformations affines (voir section 4.1). La forme en plan et la distribution de torsion sont donc requises dès la première étape de

construction.

En fonction de la corde apparente et de l'angle d'incidence, la corde réelle est trouvée, comme précédemment. Avec cette dernière, les facteurs d'échelle à appliquer aux paramètres positions et courbure des points clés sont trouvés. L'angle de torsion vient corriger les paramètres angulaires (les angles des pentes des points clés). Les amplitudes demeurent inchangées. Quant à la position du bord d'attaque, elle sert à déterminer la translation à appliquer à la station.

Les points clés sont donc construits à partir des paramètres adimensionnels, de la forme en plan et de transformations géométriques. Une courbe est passée à travers ces points pour constituer une station guide. L'ensemble de ces opérations est répété pour les trois stations guides. La figure 4.5 montre le résultat à cette étape.

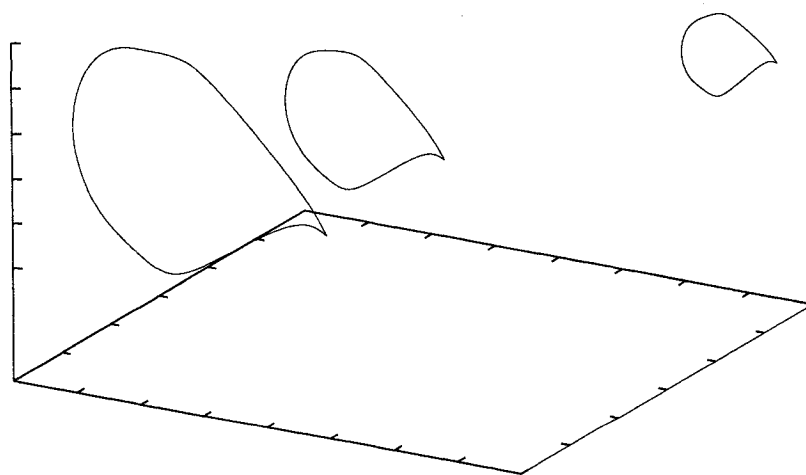


FIG. 4.5: Construction de l'aile : Construction des stations guides

4.2.3 Construction des bec et volet

Les trois profils générateurs sont maintenant construits. L'étape suivante est la construction des bec et volet et il est requis que ceux-ci soient des surfaces réglées. Étant donné la géométrie de la forme en plan, nous considérons l'aile en deux parties : la partie interne (*in-board*—de l'emplanture à la cassure) et la partie externe (*out-board*—de la cassure au bout). De plus, nous divisons les surfaces selon qu'elles fassent parties de l'intrados ou de l'extrados. Le bec se retrouve donc divisé en deux par la ligne du bord d'attaque. Avec ces considérations, les bec et volet sont constitués de quatre surfaces chacun ([intrados/extrados] \times [interne/externe] = $2 \times 2 = 4$).

Pour éviter les incohérences, la position x des bec et volets, tout comme celle des longerons, est toujours extraite de la forme en plan. Les autres caractéristiques proviennent des points clés. Si la couche de contrainte contrôlant les bec et volet est activée, la recherche des contraintes géométriques des points de recouvrement sur les stations guides est triviale. Dans le cas contraire, les emplacements des lignes de recouvrement des becs et volets issues de la forme en plan sont nécessaire pour trouver ces positions. Comme seule la position x est connue, nous devons trouver les paramètres d'avancement u correspondant aux points de recouvrement.

Une fois ces quatre points trouvés sur une station, cette dernière est coupée pour ne conserver que les segments correspondant aux bec et volet. Un nouveau vecteur nodal normalisé ($0 \leq u \leq 1$) est construit sur ces portions de courbes. Cette opération est répétée pour toutes les stations guides. Enfin les huit surfaces réglées sont construites à partir de ces courbes.

La ligne de bord d'attaque de l'aile naît de la jonction de la portion sur l'intrados et l'extrados du bec des stations guides. Il en est de même pour la ou les (s'il est ouvert) ligne(s) de bord de fuite. Une ligne de cassure est également visible sur le bec et sur le

volet à la jonction des surfaces intérieures et extérieures. La figure 4.6 illustre l'aile à cette étape de construction.

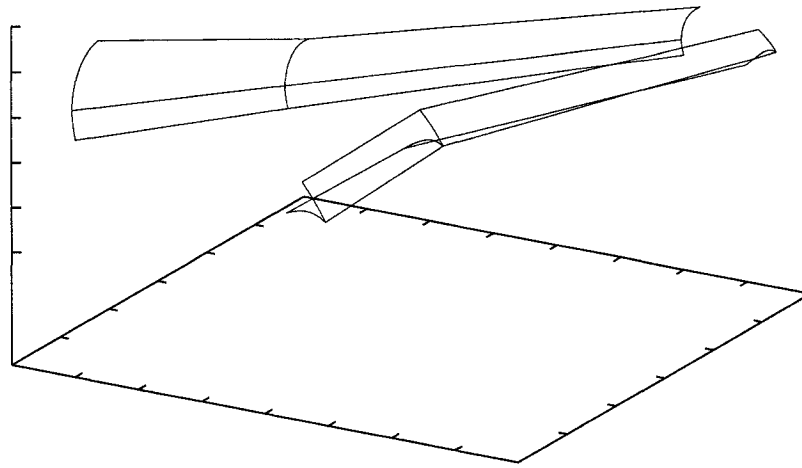


FIG. 4.6: Construction de l'aile : Construction des bec et volet

Cependant, il est possible de vouloir pour le bec, dans certaines situations, une surface réglée de l'emplanture au bout, donc sans cassure. Dans ces circonstances, il est possible d'omettre une station guide et de générer la surface du bec sur toute l'envergure avec les deux autres. Les points clés correspondants de la station omise sont simplement ignorés et le programme vient créer de nouvelles courbes représentant le bec à la bonne station à l'aide de l'interpolation ou de l'extrapolation (projection) des deux autres. Nous nous trouvons alors à imposer une variation linéaire de l'élévation z du bec en fonction de la station, le bec devient une surface lisse sans cassure sur toute l'envergure.

Il est également possible d'imposer une variation linéaire de l'élévation z pour le volet, bien que dans ce cas la cassure ne peut être éliminée, à cause de la forme en plan. La

même stratégie d'interpolation ou de projection est employée, mais suivie d'une translation pour respecter la forme en plan. Si la même station guide est omise pour le bec et le volet, elle n'est pas générée lors de la première étape (section 4.2.2). C'est pour cette raison, que parfois seules deux stations guides sont construites.

Notons aussi la possibilité de fusionner les points clés 1-3 et 2-3 (longeron avant et bec sur l'extrados). Cette situation est rencontrée lorsque les paramètres de ces derniers sont identiques sur la forme en plans. Cette situation est fréquente avec l'emploi de bec de type Kruger. Dans ce cas, la position du longeron avant sur l'extrados suit l'interpolation linéaire prescrite par le bec.

4.2.4 Construction des surfaces intérieures

Il nous faut maintenant générer les courbes représentant les surfaces intérieures, *i.e.*, entre le bec et le volet sur l'intrados et l'extrados. Le tout devra être continu en courbure dans le sens de la corde (aux quatre jonctions des bec et volet) et dans le sens de l'envergure (à la cassure). Comme les analyses que nous allons effectuer sur l'aile ont besoin de discrétisations sur certaines stations de celle-ci, la méthode proposée extrait la courbe correspondant à la surface à toute station. La surface n'est donc pas disponible en tant que surface B-Spline (produit tensorielle), comme les bec et volet, mais comme une fonction donnant la géométrie d'une station en fonction de l'envergure. La position des stations désirées est donnée dans le fichier de construction.

Pour nous assurer de la continuité et de la régularité dans le sens de l'envergure, des interpolants coniques sont construits sur chaque paramètre dimensionné qui a de l'influence sur la partie intérieure. Le nombre de paramètres varie en fonction du nombre de couches activées. Ainsi, les paramètres relatifs au bord d'attaque et de fuite n'ont pas d'influence dans les parties intérieures de l'intrados et de l'extrados, car leur influence

affecte seulement les becs et volets.

Sur chaque station désirée, il nous faut construire deux nouvelles B-Splines pour constituer l'intrados, entre le volet et le bec, et l'extrados, entre le bec et le volet. Ces splines sont régies par un sous-ensemble des contraintes que nous avons vues jusqu'à présent. Il faut donc extraire, à l'aide des interpolants coniques, les valeurs des paramètres affectés aux stations voulues. Les points clés sont construits avec ces valeurs. De plus, à l'aide du bec et du volet, pour ces mêmes stations, les valeurs de positions, pentes et courbure à imposer pour assurer la continuité aux points de jonction sont extraites. Ces valeurs servent à générer les points clés aux extrémités des courbes intérieures. La figure 4.7 montre l'aile avec les station guides, les bec et volet et la partie intérieure d'une station sur la partie externe de l'aile. Nous voyons aussi sur cette figure les interpolants des positions maximales et minimales, qui correspondent au squelette de contraintes.

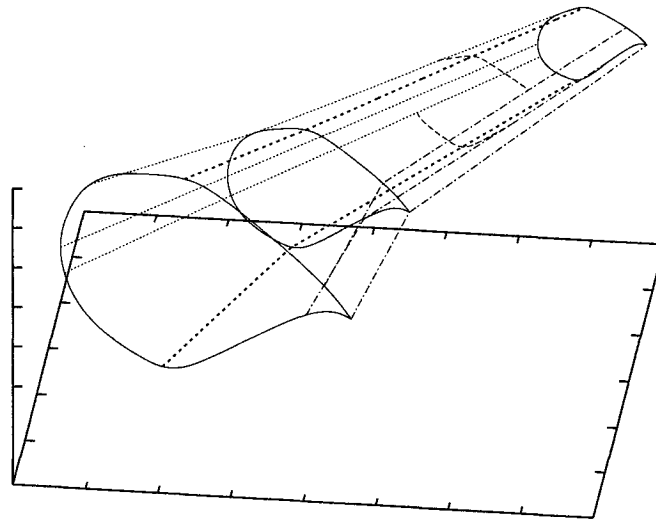


FIG. 4.7: Construction de l'aile : Construction des surfaces intérieures

Comme aucune transformation affine n'est nécessaire après génération d'un profil, la continuité à la cassure est préservée. De plus, en utilisant les paramètres dimensionnés sur les stations, la ligne du bord d'attaque est conservée droite et la tangente à ce point est bien verticale. Les défauts inhérents au premier module sont donc évités.

À l'aide de cette façon de construire l'aile, la surface de l'aile est connue aux endroits désirés. Il faut maintenant la rendre plus cohérente, en ce sens que les entités sont fournies parfois par produit tensoriel, comme les bec et volet, alors que le reste l'est sous forme de plusieurs courbes paramétriques. Il faut donc extraire les courbes des bec et volet aux stations désirées pour les joindre à celles créées à l'étape précédente.

4.2.5 Discrétisation

Nous emploierons la courbure comme base de discrétisation des stations. Nous voulons davantage de points discrets là où il y a plus de variation (où la courbure est plus grande). Un paramètre de décalage sur la courbure permet d'influencer la concentration des points en fonction de la courbure. Avec un paramètre de décalage fixé à zéro, seule la courbure intervient dans la distribution, posé à l'infini, les points sont tous également distribués. La figure 4.8 montre la courbure intégrée en fonction du paramètre de décalage et la figure 4.9 la distribution de points discrets résultante. Notons que sur cette dernière figure, le nombre de points de discrétisation est dérisoirement bas pour bien voir l'effet du paramètre de décalage.

Afin de générer la fonction de concentration de courbure, nous évaluons le profil à un grand nombre d'endroits h , où h est un ou deux ordres de grandeur supérieur au nombre de points discrets final voulu. Nous divisons le vecteur nodal en pas de $1/h$. Bien que les paramétrisations puissent ne pas être intrinsèques, le grand échantillonnage choisi contourne ce problème. À tous ces u_r , nous évaluons la position et la courbure afin de

constituer la fonction de concentration à l'aide de :

$$\oint \kappa(u) = \sum_{i=1}^h \left(\frac{(\kappa_i - \kappa_{i-1}) + d}{L_i} \right) \quad (4.1)$$

Où $\oint \kappa(u)$ est la fonction de concentration de la courbure intégrée discrètement selon la longueur de corde en fonction de l'avancement u , κ_i est la courbure au point u_i , d est le paramètre de décalage et L_i la longueur du segment i . Cette fonction, que l'on normalise de 0 à 1, sert de base à la sélection des points discrets.

La fonction de concentration est à son tour échantillonnée en n intervalles de grandeur constante, où n est le nombre de points discrets voulus. À ces n valeurs sont associées un point sur le profil, les points de discrétisation choisis.

L'équation 4.1 permet de bien voir l'effet du paramètre de décalage. S'il est très grand, il domine le numérateur, et l'évolution de $\oint \kappa$ est constante, d'où une répartition égale des points discrets. Dans le cas contraire, s'il est très petit, son effet est négligeable et les points discrets suivent l'évolution de la courbure. Enfin, pour les valeurs intermédiaires, le paramètre de décalage gouverne la discrétisation où la courbure est faible (près des points d'inflexion, minimum et maximum), mais de grandes courbures, comme au bord d'attaque, engendre des points discrets supplémentaires.

Un paramètre de décalage entre 3 et 6 pour une centaine de points de discrétisation permet d'obtenir un profil où l'erreur de représentation est faible. La figure 4.10 montre la discrétisation de l'aile ainsi obtenue définie sur 8 stations avec un décalage de 6. Le paramètre de décalage et le nombre de point de discrétisation sont fournis dans le fichier de construction. Le module permet quelques variantes dans la discrétisation. Il peut intégrer la courbure sur tout le profil, par coté (extrados et intrados) ou par section (bec, volet, parties intérieures). L'avantage d'intégrer la courbure en plusieurs morceaux sur le profil est qu'il y ait toujours un point à la jonction de ces morceaux. Ainsi, en

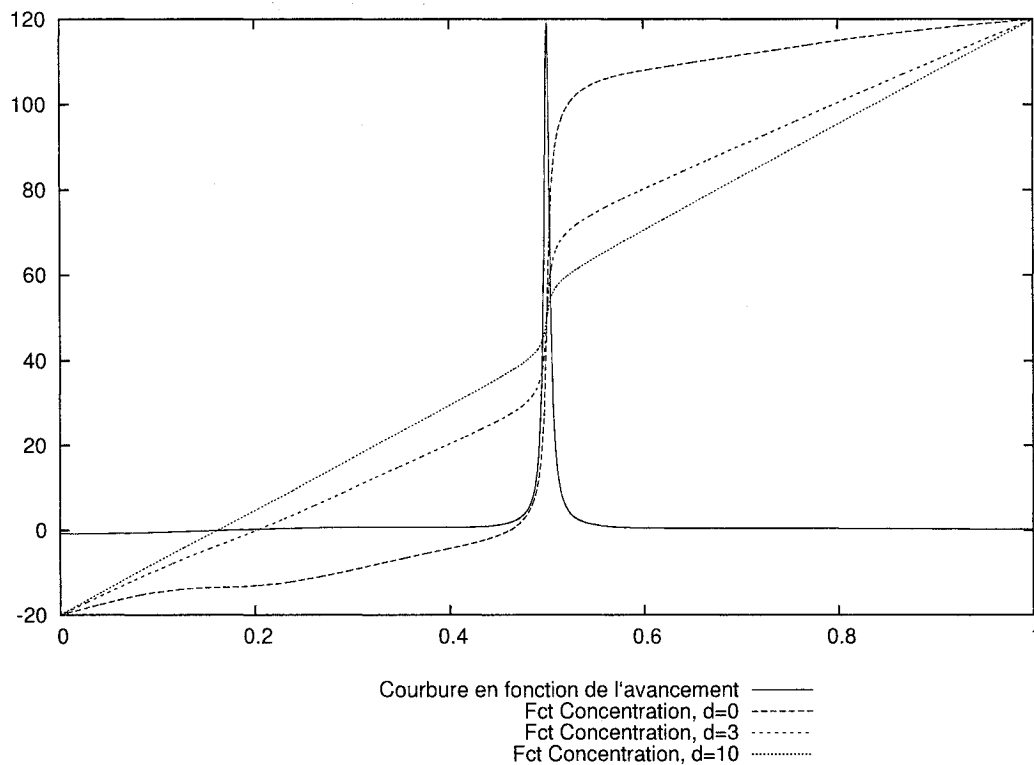


FIG. 4.8: Effet du paramètre de décalage sur la fonction de concentration de la courbure d'un profil

procédant par coté (intrados/extrados), il y a toujours un point au bord d'attaque et par section il y en a toujours sur les lignes de recouvrement.

Nous avons remarquer que dans certains cas de constructions d'aile, la combinaison des valeurs des caractéristiques géométriques aux points clés ainsi que l'enveloppe de la forme en plan amènent à générer des oscillations. Celles-ci proviennent du fait que des points clés consécutif rapprochés ont de grandes différences de position, de pentes ou de courbure. De plus, quand les lignes de recouvrement sont très rapprochées des longerons, certaines oscillations peuvent apparaître. Celle-ci naissent de la différence entre ces deux entités : la première est générée linéairement alors que la seconde l'est selon une conique. Pour éviter que des oscillations apparaissent, le choix des paramètres géométriques est donc crucial.

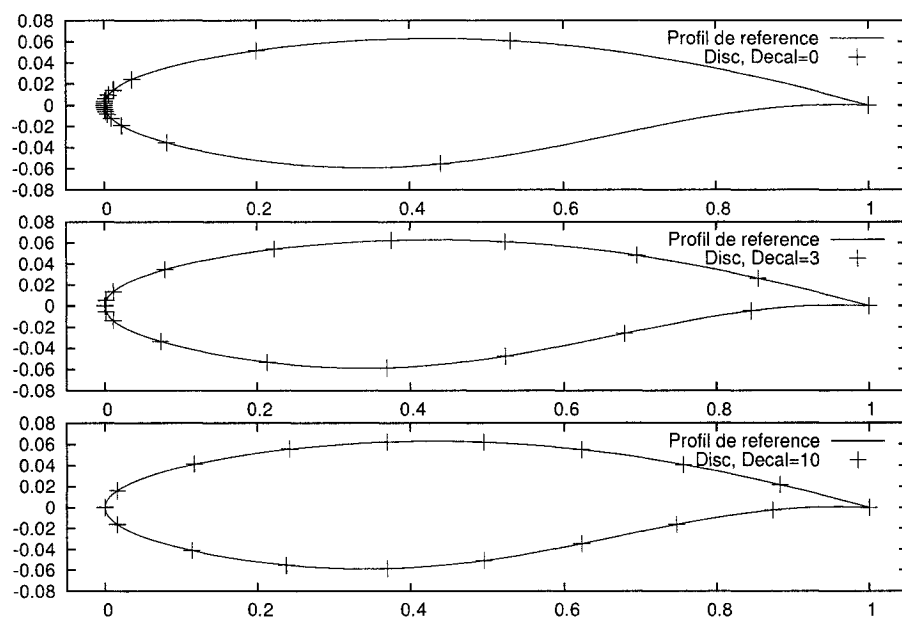


FIG. 4.9: Effet du paramètre de décalage sur la distribution des point discrets sur un profil

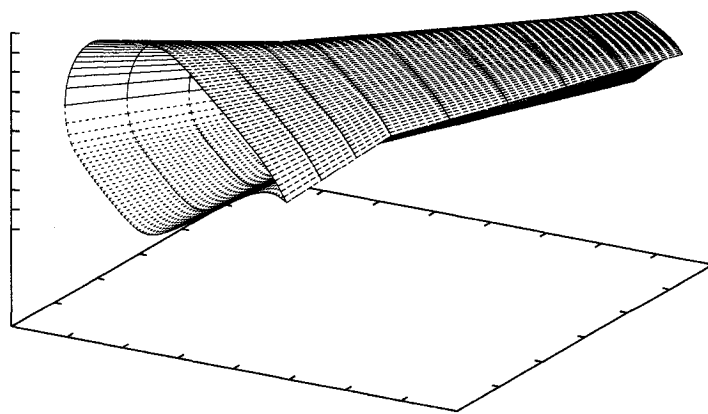


FIG. 4.10: Construction de l'aile : Discrétisation

CHAPITRE 5

VALIDATION GÉOMÉTRIQUE

MAINTEANT que nous avons développé une paramétrisation s'appuyant sur des courbes géométriquement contraintes qui permet la représentation 3D d'une aile, il reste à tester les possibilités et limites de cette représentation. Pour ce faire, nous procéderons en deux étapes. Dans un premier temps, nous tenterons de représenter des profils aérodynamiques adimensionnels connus à l'aide des méthodes d'interpolation présentées au chapitre 3. Ensuite, la méthode de génération d'une aile transsonique par stations présentée au chapitre 4 est utilisée afin de répéter la même expérience sur une aile tridimensionnelle typique.

5.1 Formulation

Dans les deux cas, il s'agit de trouver le vecteur de paramètres, ou variables de design, représentant le mieux la géométrie cible. Autrement dit, nous procédons à une optimisation géométrique de profils et d'une aile. La fonction coût $f(\mathbf{x})$ utilisée est une mesure de l'erreur de représentation ϵ entre la géométrie paramétrée et la géométrie cible. Le vecteur de design \mathbf{x} est constitué d'un sous-ensemble des paramètres d'ingénierie décrits dans les chapitres précédents, car certains paramètres n'ont pas à être optimisés comme l'angle du bord d'attaque par exemple. Le vecteur de design a une dimension n variable allant d'une dizaine de paramètres à plus d'une centaine en fonction de l'environnement d'optimisation (profil ou aile, couches activées, paramètres sélectionnés, etc.). Toutes les optimisations menées ont comme seules contraintes les bornes à appliquer x_i^L et x_i^U aux variables de design, car le seul critère est la qualité de la représentation géométrique, ou

fonction coût. La méthode quasi-Newton BFGS (voir II.2) est utilisée afin de mener ces optimisations. La formulation générale du problème est donnée par :

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}) \\ \text{tel que :} \quad & x_i^L \leq x_i \leq x_i^U, \quad 0 \leq i < n \end{aligned} \quad (5.1)$$

Notons que les paramètres à optimiser sont adimensionnalisés dans une plage allant de -1 à 1 à partir de la plage de contrainte dimensionnelle qui leur est allouée afin qu'ils aient tous le même ordre de grandeur. En notant \tilde{x} une variable dimensionnelle et x une variable adimensionnelle, le passage de l'un à l'autre s'effectue comme suit :

$$x = 2 \left(\frac{\tilde{x} - \tilde{x}^L}{\tilde{x}^U - \tilde{x}^L} \right) - 1. \quad (5.2)$$

et donc $x_i^L = -1$ et $x_i^U = 1$ quelque soit i .

5.2 Validation géométrique sur des profils 2D

Dans un premier temps, la mise en place du problème d'optimisation est présentée. Ensuite, les deux méthodes de génération de courbes présentées au chapitre 3 sont comparées, suivies d'une comparaison de deux stratégies d'optimisation ; la première a comme seules variables de design les amplitudes aux points clés alors que la seconde optimise tous les paramètres pouvant varier. Enfin, vient une étude de la précision en fonction du nombre de couches activées avant de conclure sur l'optimisation géométrique bidimensionnelle.

5.2.1 Mise en place du problème d'optimisation bidimensionnel

Dans le cas de profils 2D, les deux méthodes d'interpolation présentées au chapitre 3 sont évaluées afin de déterminer laquelle est la mieux adaptée pour la représentation de profils. Les deux méthodes, celle par collage de B-Splines contraintes aux extrémités et celle d'une B-Spline multi-contraintes en plusieurs endroits, sont dorénavant respectivement appelées méthode #1 et #2.

Pour valider que le spectre de la paramétrisation est suffisamment large, nous avons procédé à l'optimisation géométrique de six profils différents, dont voici la liste : RAE2822 (*Royal Aircraft Establishment [UK]*), bacj (*Boeing Commercial Airplane Company airfoil J*), Whitcomb (*NASA/Langley Whitcomb integral supercritical airfoil*), npl9510 (*National Physical Laboratory [UK]*), K2 (*Grumman transonic airfoil*) et BAA (*Bombardier Aeronautic Airfoil*). Ces profils ont été retenus car ce sont des profils supercritiques fréquemment utilisés comme cas tests et/ou sur des aéronefs évoluant en régime transsonique.

La fonction coût utilisée pour mener les optimisations géométriques mesure l'erreur de représentation entre deux profils. Toutes les optimisations géométriques ont lieu sur des profils adimensionnels, *i.e.* dont la corde $c = 1$. Comme les profils cibles sont donnés sous forme de listes de points, une B-Spline les interpolant est construite afin d'avoir une représentation continue à tout endroit de ceux-ci. La mesure de l'erreur de représentation se base sur la distance euclidienne entre le profil paramétré et le profil B-Spline cible. Il est possible de calculer la distance d'un point appartenant à une courbe B-Spline sur une autre B-Spline à l'aide d'algorithmes de projection et d'inversion (Piegl et Tiller 1995). La distance est extraite à un grand nombre d'endroits à partir du profil paramétré afin d'obtenir une mesure représentative de l'erreur.

La mesure de l'erreur s'appuie sur deux termes : la distance moyenne et la distance

maximum. Si seule la distance maximale est retenue, l'optimiseur a tendance à n'améliorer qu'un endroit du profil à la fois : la région où la distance est maximale. Par contre, si seule la distance moyenne est conservée, l'endroit où la représentation est la pire est estompé dans la fonction coût et les optimisations sont lentes. Lépine (1999) a constaté que la distance maximale est généralement environ deux fois plus grande que la distance moyenne. Pour conserver les mêmes ordres de grandeur pour ces deux termes afin que la fonction coût soit cohérente, elle est exprimée comme suit :

$$f(\mathbf{x}) = 2\epsilon_{moy}(\mathbf{x}) + \epsilon_{max}(\mathbf{x}) \quad (5.3)$$

avec

$$\begin{aligned} \epsilon_{moy} &= \frac{1}{n} \sum_{i=1}^n \epsilon_i(\mathbf{x}) \\ \epsilon_{max} &= \max |\epsilon_i| \end{aligned} \quad (5.4)$$

où n est le nombre de mesures et ϵ_i une mesure de distance. Au delà de 500 évaluations de la distance, les deux termes constituant la fonction coût cessent de varier significativement, c'est donc ce nombre de mesure qui sera utilisé. Notons que ces mesures sont prises à des longueurs équidistantes sur le profil paramétré et non pas en les concentrant en fonction de la courbure.

Pour ces validations géométriques, l'ordre des points clés est mis à deux, *i.e.*, la position, la pente, l'amplitude et la courbure servent à construire les profils à tous les points clés. Cependant, ce ne sont pas nécessairement tous les paramètres qui sont optimisés, certains sont imposés à des valeurs prescrites.

Pour les six profils retenus, les valeurs de bornes des paramètres demeurent inchangés aux valeurs indiqués au tableau 5.1 à l'exception de l'amplitude dont la plage est moins grande pour la méthode #1 ([0.1, 2]) que pour la méthode #2 ([0.1, 5]). S'il n'y a pas de plage donnée avec le paramètre, c'est que celui-ci est imposé à la valeur donnée et n'est pas optimisé.

TAB. 5.1: Bornes ou valeurs des paramètres des points clés pour la validation 2D

Couche	Nom PC	Pos _x	Pos _z	Angle	Courbure	Amplitude
0	TE _{in}	1	0	[150, 180]	[0, 6]	[0.1, 2/5]
	IN	[0.25, 0.4]	[-0.07, -0.004]	180	[-1, -0.2]	[0.1, 2/5]
	LE	0	0	90	[-150, -35]	[0.1, 2/5]
	EX	[0.3, 0.55]	[0.045, 0.065]	0	[-0.5, -0.1]	[0.1, 2/5]
	TE _{in}	0	1	[-25, -5]	[-12, 0]	[0.1, 2/5]
1	RSI	0.65	[-0.035, -0.02]	[181, 199]	[-2, 1]	[0.1, 2/5]
	FSI	0.15	[-0.06, -0.03]	[167, 179]	[-1.5, -0.2]	[0.1, 2/5]
	FSE	0.15	[0.03, 0.06]	[2, 12]	[-1.5, -0.2]	[0.1, 2/5]
	RSE	0.65	[0.03, 0.06]	[-10, 0]	[-1, 0]	[0.1, 2/5]
2	FI	0.75	[-0.04, 0]	[181, 196]	[-1, 1.5]	[0.1, 2/5]
	SI	0.05	[-0.045, -0.015]	[155, 180]	[-4.5, -0.5]	[0.1, 2/5]
	SE	0.12	[0.03, 0.05]	[2, 12]	[-1.6, -0.1]	[0.1, 2/5]
	FE	0.85	[0.015, 0.045]	[-12, 2]	[-1, 0]	[0.1, 2/5]
3	Cavité	[0.85, 0.99]	[-0.002, 0.02]	180	[0.5, 5.5]	[0.1, 2/5]
	Inflexion	[0.55, 0.95]	[-0.04, 0]	[181, 200]	0	[0.1, 2/5]
4	SSI	0.7	[-0.03, -0.01]	[181, 200]	[-1, 2]	[0.1, 2/5]
	SSE	0.7	[0.03, 0.055]	[-10, 0]	[-1, 0]	[0.1, 2/5]

Comme certaines couches s'appuient sur des caractéristiques structurales, comme les couches un, deux et quatre, plutôt que sur des caractéristiques géométriques, comme la couche trois, nous devons explicitement fixer les emplacements x en pourcentage de corde des longerons et des points de recouvrement des bec et volet. Pour ce faire, des valeurs typiques sont employées et celles-ci demeurent fixes pour toute la durée des optimisations afin que les points clés restent aux emplacements x prescrits. Ainsi, les longerons avant et arrière de la couche numéro un sont positionnés respectivement à 15 et 65 % de corde et celui de la couche quatre à 70 %. Les points de recouvrement du bec sont pour l'intrados et l'extrados, situé à 5 et à 12 % de corde, et pour le volet, à 75 et 85 %.

Afin de déterminer le point de départ (vecteur de design initial) pour l'optimisation des différents profils, les B-Splines interpolant les points des profils cibles ont été utilisées. À partir de celles-ci, les valeurs de position, de pente et de courbure ont été extraites là où

nécessaire et ont servi afin de générer le premier profil paramétré. Pour la méthode #1, toutes les amplitudes sont mises à l'unité et le profil initial représente bien le profil cible. Pour la méthode #2, les amplitudes mises à l'unité ne nous assurent pas un profil initial aussi près qu'avec la méthode #1 (voir l'exemple de la section 3.1.4 qui montre également ce phénomène). Les amplitudes doivent donc être ajustées manuellement sur les profils initiaux afin d'obtenir une meilleure représentation pour initialiser l'optimisation (plus de détails sont disponibles à la section suivante).

Plusieurs optimisations ont été menées sur les six profils de l'étude avec différents pas de différenciation afin de voir l'effet qu'à ce dernier sur les résultats des optimisations. Nous voyons sur la figure 5.1 que les meilleurs résultats sont généralement obtenus pour un pas de différenciation autour de 10^{-3} - 10^{-4} . Les optimisations seront donc conduites en double, une pour chaque pas afin de sélectionner celle qui donne de meilleurs résultats. Il a été remarqué qu'un pas de 10^{-4} sur les variables de design adimensionnelles donne généralement les représentations les plus justes.

5.2.2 Comparaison des deux méthodes de génération des B-splines

Dans cette section, nous comparons les deux méthodes de construction des B-Splines contraintes géométriquement proposées au chapitre 3. Une série d'optimisation a été menée pour les deux méthodes de construction. La robustesse des méthodes ainsi que la qualité des représentations seront comparées.

Comme mentionné plus haut, un problème rencontré avec la méthode #2 concerne l'ajustement des amplitudes des dérivées premières. En effet, en les laissant à l'unité, la courbe construite est relativement loin du profil cible entre certains points clés, comme le montre l'exemple du profil rae2822 sur la figure 5.2(a). Sur cette figure, la grandeur de la région d'influence des propriétés géométriques (voir explication de la nature de l'amplitude à la

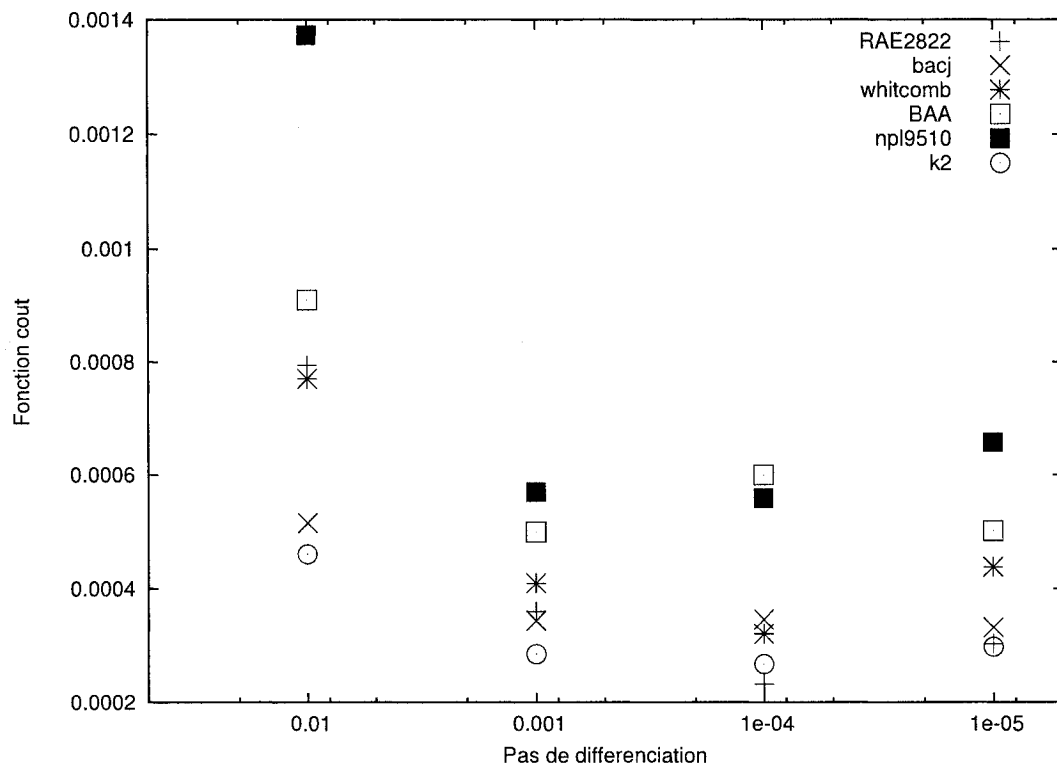
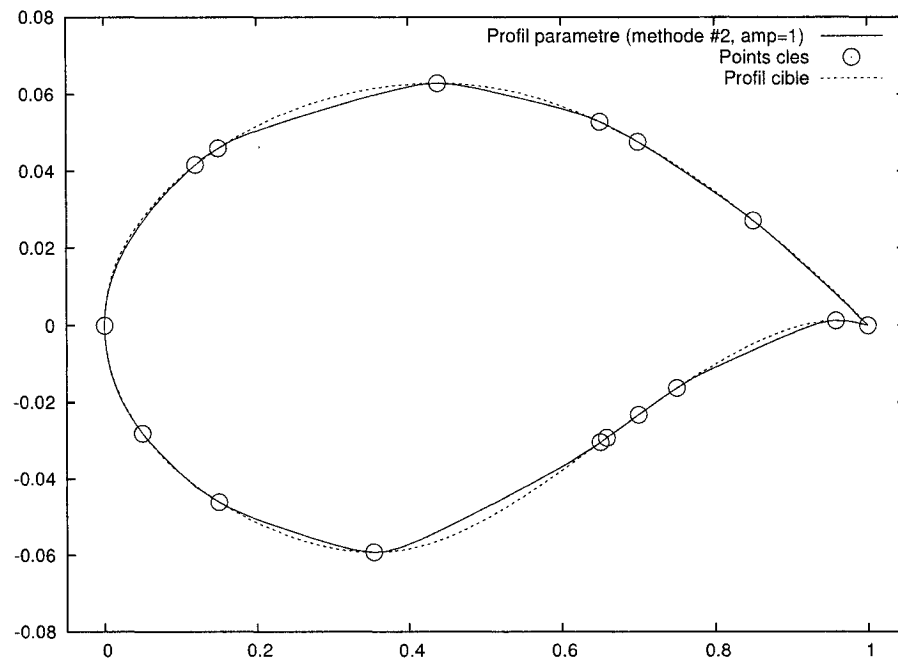


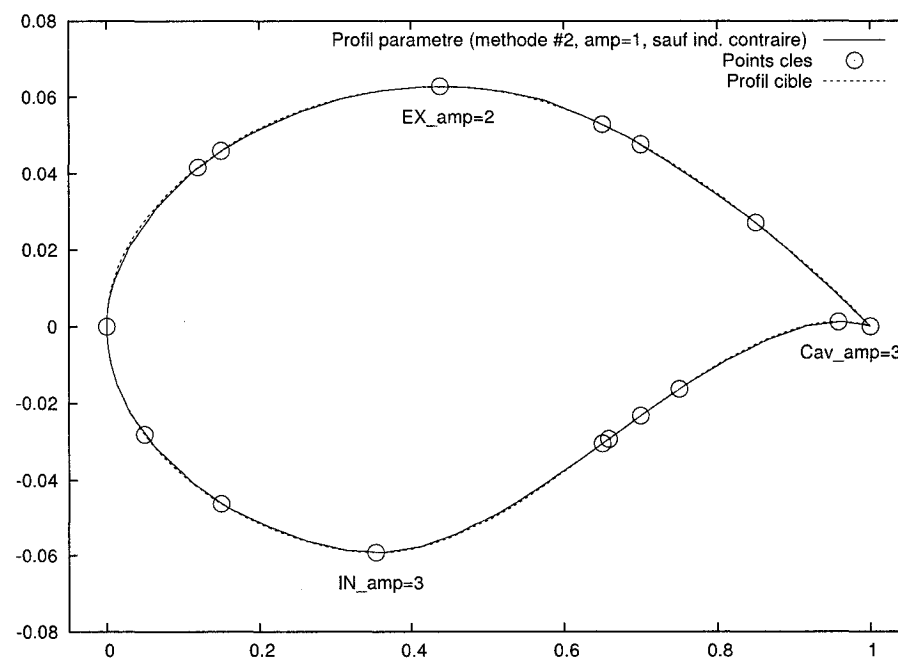
FIG. 5.1: Effet du pas de différenciation sur la fonction coût de différents profils

page 60) imposées aux points clés est fréquemment insuffisante, ce qui donne l'impression que des lignes droites interpolent la majeure partie de l'espace entre certains points clés, notamment autour des points maximum et minimum et de la cavité.

Un ajustement des amplitudes est nécessaire afin d'améliorer la représentation constituant le point de départ des optimisations. Ainsi, en augmentant la valeur des amplitude aux points clés, là où leur influence n'est pas assez grande, il est possible, en quelques essais, d'obtenir une représentation satisfaisante comme design initial. Pour l'exemple des figures 5.2, l'amplitude des points clés correspondant à la cavité ainsi que celle des points maximum et minimum, à été augmentée de 1 à respectivement 3, 4 et 3. Le résultat, beaucoup plus proche du profil cible, est illustré à la figure 5.2(b). Cependant, ajuster manuellement les amplitudes, en plus de rendre le processus tributaire d'une intervention humaine, ne règle pas tous les problèmes. En effet, si certaines couches sont désactivées, il se peut que la représentation du profil paramétré comporte les mêmes problèmes que



(a) Profil où toutes les amplitudes sont laissées à l'unité



(b) Profil où certaines amplitudes ont été ajustées pour permettre une meilleure représentation

FIG. 5.2: Illustration de l'effet de l'ajustement des amplitudes sur un profil construit à l'aide de la méthode #2 où toutes les autres caractéristiques géométriques sont fixes

ceux illustrés à la figure 5.2(a), un peu comme si une certaine combinaison de couches actives devait être utilisée avec un certain jeu de valeur d'amplitude. Mais comme un seul point de départ a été utilisé, il n'y a pas eu d'autres ajustements des amplitudes. Enfin, les cinq autres profils ont aussi vu leurs amplitudes initiales ajustées manuellement afin d'obtenir un bon point de départ avec toutes les couches activées. Comme cet ajustement n'est nécessaire que pour la deuxième méthode, la première est donc plus versatile, plus facilement utilisable et automatisable.

Côté robustesse, la méthode #1 semble mieux adaptée au type de problème d'optimisation à l'étude. Au cours des différentes optimisations menées, la méthode #1 a toujours bien fonctionné, alors qu'il est arrivé à plusieurs occasions que celles conduites avec la méthode #2 s'arrêtent rapidement (à peine une itération), voire qu'elles ne réussissent pas à trouver de direction de descente pour initialiser le calcul (zéro itération). Ceci peut s'expliquer par le fait que le point de départ des optimisations avec la méthode #2 avec quelques couches activées était parfois loin du profil cible. De plus, il a été constaté qu'avec certaines combinaisons de valeur d'amplitudes, les profils construits comportent des points de rebroussement ou se recoupent en raison d'un mauvais vecteur nodal, donc d'une mauvaise distribution des points de contrôle. Ce type de problème interrompt les optimisations, car la géométrie paramétrée ne représente plus un profil aérodynamique et donc, la fonction coût n'est plus valable. Cette grande sensibilité, qui rend la deuxième méthode moins robuste, la rend moins attrayante, car cela compromet parfois les optimisations.

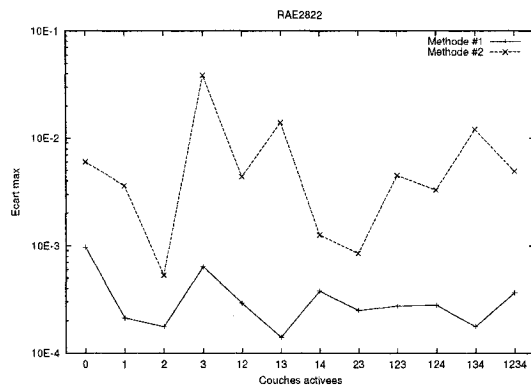
Côté précision, nous pouvons généralement affirmer que la méthode #1 fournit de meilleurs résultats. En effet, sur toutes les optimisations menées, dans à peine plus de 10% des cas la deuxième méthode a donné de meilleures représentations. De ces meilleurs cas, la grande majorité a été obtenue sur les optimisations avec amplitudes seulement. La figure 5.3 illustre les écarts maximum des optimisations complètes des différents profils en fonction de la méthode de construction. Nous y voyons que les précisions obtenues

avec la méthode #1 sont généralement un ordre de grandeur inférieur à celles obtenues avec la méthode #2.

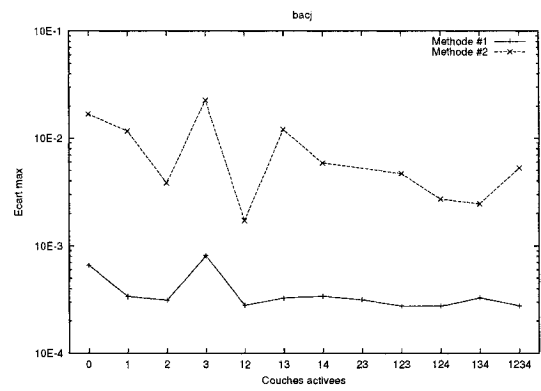
Comme les mathématiques pour imposer les caractéristiques géométriques à la base des deux méthodes de construction des B-Splines sont relativement semblables, les meilleures performances de la méthode #1 peuvent s'expliquer par la génération d'un meilleur vecteur nodal. En effet, comme nous le mentionnions au chapitre 3, il est difficile de générer un bon vecteur nodal pour une courbe avec contraintes d'ordre $k = 2$ (position, pente, amplitude et courbure). La méthode #1 contourne ce problème en construisant par pièce la courbe finale, et c'est dans l'opération de collage qu'est générée automatiquement le vecteur nodal final. Dans chaque morceau de courbe, les vecteurs nodaux sont plus faciles à générer, car seulement deux points clés interviennent. Pour la méthode #2, le vecteur nodal, généré de façon semi-empirique, doit être connu afin d'assembler la matrice pour positionner les points de contrôle. Il apparaît donc que la façon d'y parvenir peut sans doute être améliorée. Pour l'ensemble de ces raisons, la méthode #1 est jugée plus adéquate à la construction de profils aérodynamiques. Comme elle est également plus robuste, cette méthode est utilisée par défaut et sera employée pour la génération de l'aile à la section suivante.

5.2.3 Comparaison des optimisations avec et sans amplitude

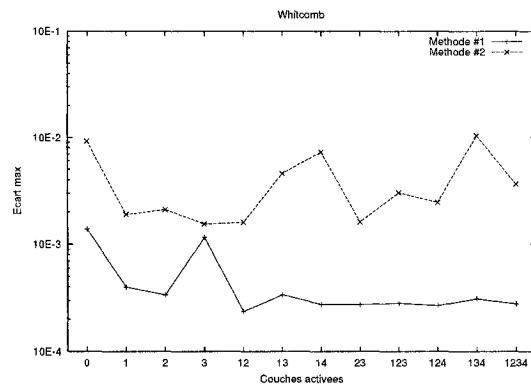
À présent, la différence entre les optimisations où seules les amplitudes sont laissées libres versus celles où tous les paramètres disponibles peuvent varier est étudiée. La paramétrisation développée s'appuie sur des paramètres géométriques. Donc, théoriquement, comme les points de départ des différentes optimisations sont issus de l'extraction des caractéristiques géométriques sur des B-splines représentant les profils cibles, les profils paramétrés devront être pratiquement superposés à ceux-ci. Ceci est illustré sur la figure 5.4 ($F_{\text{coût init}} = 1.88\text{E-}3$). Il est donc possible de n'optimiser que les amplitudes et



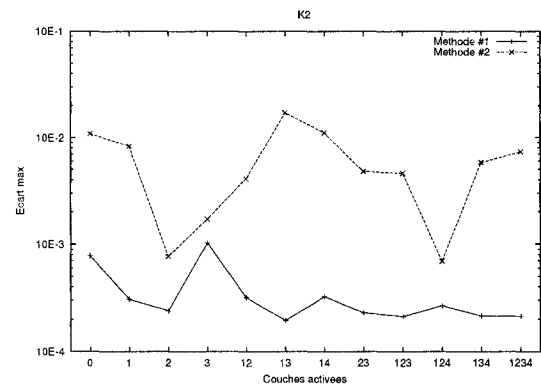
(a) RAE2822



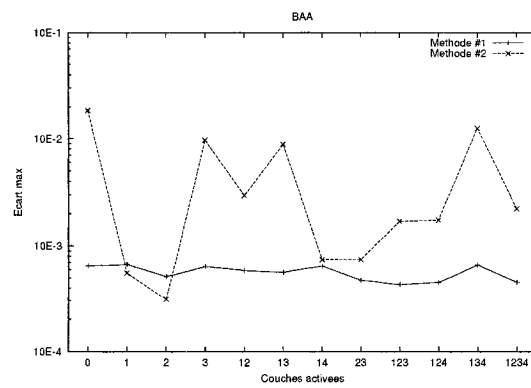
(b) bacj



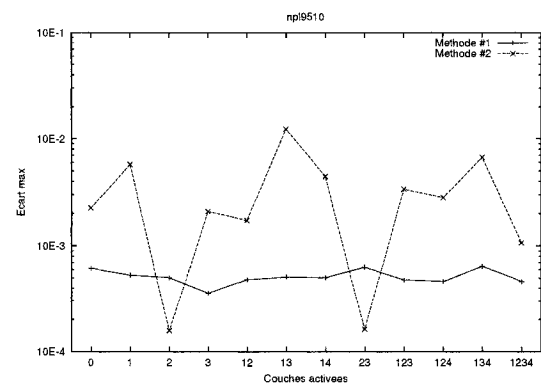
(c) Whitcomb



(d) k2



(e) BAA



(f) npl9510

FIG. 5.3: Comparaison des écarts maximum entre les deux méthodes de construction des profils en fonction des caractéristiques de l'optimisation (trait plein : méthode #1, tireté : méthode #2)

de s'attendre à ce que la représentation s'approche grandement du profil cible, ce qui a été obtenu (figure 5.5(a), $F_{\text{cout}} = 2.65\text{E-}4$). Cependant, les meilleurs résultats s'obtiennent tout de même lorsque tous les paramètres sont libres (figure 5.5(b), $F_{\text{cout}} = 1.04\text{E-}4$).

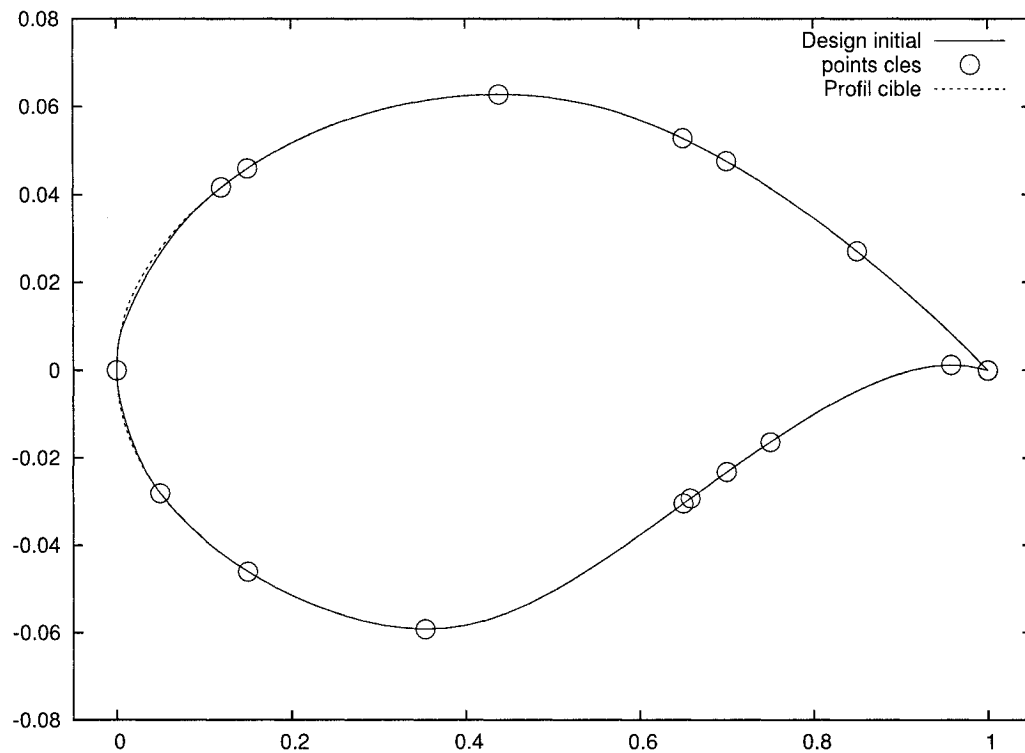
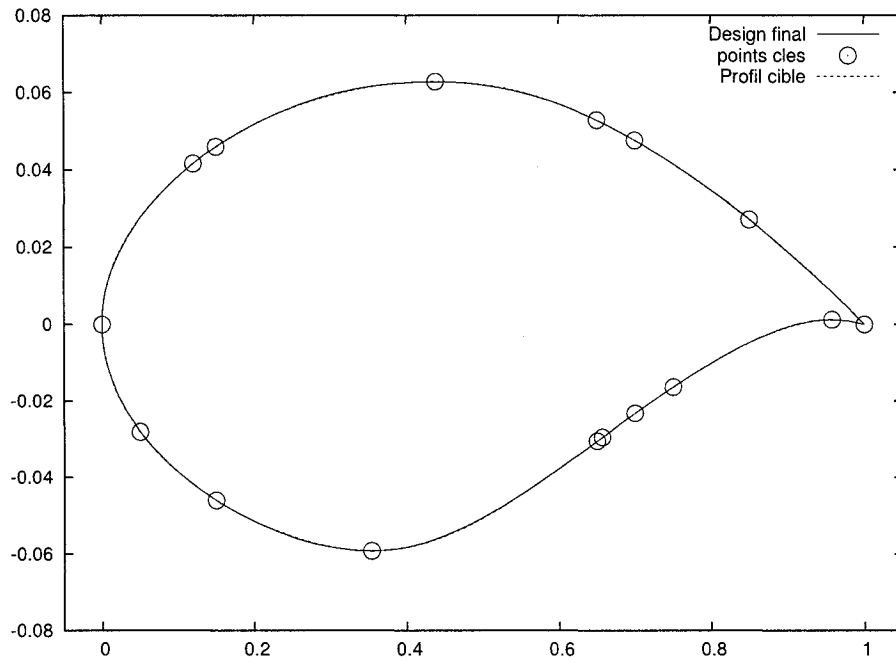
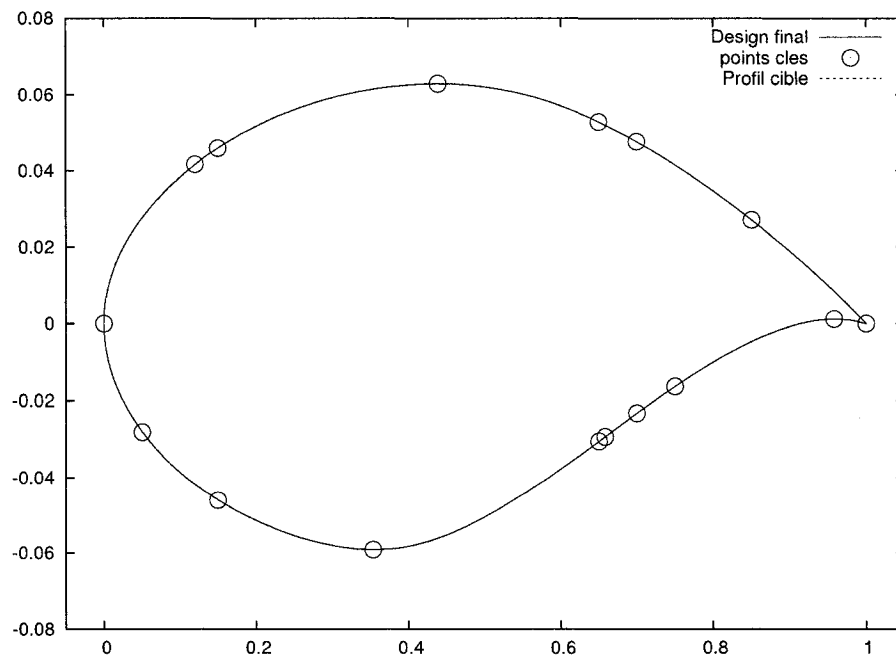


FIG. 5.4: Design initial pour la validation géométrique du profil rae2822 ; $F_{\text{cout}}=1.88\text{E-}3$, $E_{\text{max}}=1.75\text{E-}3$

La raison pour laquelle de meilleures représentations sont trouvées en laissant les paramètres géométriques obtenus à partir des profils cibles varier par rapport à lorsqu'ils sont fixés est que le point de départ obtenu ne représente exactement le profil cible qu'aux points clés. L'interpolation entre ceux-ci se rapproche grandement de la cible, mais n'est pas directement sur celle-ci. Alors, l'optimiseur, est en mesure de trouver de meilleurs compromis sur tout le profil en faisant varier un peu ces paramètres géométriques. Bien que ces variations soient très faibles (généralement inférieur à 2%), elles sont suffisantes pour que de meilleures représentations soient trouvées. En contrepartie, les durées des optimisations avec tous les paramètres sont beaucoup plus longues que celles où seules les amplitudes sont optimisées.



(a) Design final avec optimisation des amplitudes seulement ; $F_{\text{cout}}=2.65\text{E-}4$, $E_{\text{max}}=1.71\text{E-}4$



(b) Design final avec tous les paramètres disponibles ; $F_{\text{cout}}=1.04\text{E-}4$, $E_{\text{max}}=7.64\text{E-}5$

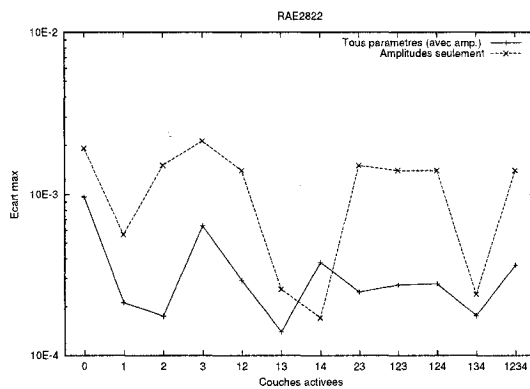
FIG. 5.5: Design final avec amplitude et avec tous les paramètres pour le profil rae2822

Enfin, la figure 5.6 illustre les écarts maximum obtenus selon les paramètres optimisés pour les différents profils. Nous voyons sur cette figure que les écarts maximum des optimisations avec amplitudes sont toujours plus élevés que ceux où tous les paramètres sont actifs, à l'exception d'une optimisation (rae2822 avec les couches 1 et 4 actives). Comme il s'agit du seul point où tel phénomène est observé, tout porte à croire que l'optimisation avec tous les paramètres s'est arrêtée dans un minimum local et que la fonction coût est fortement non linéaire.

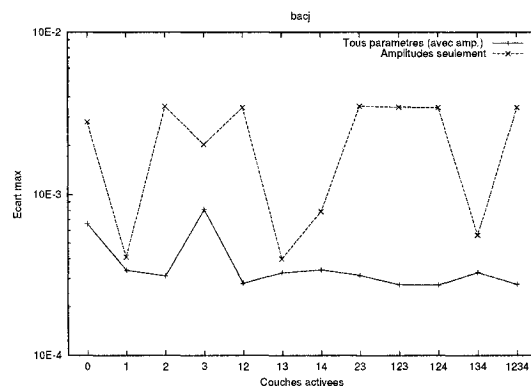
5.2.4 Effet du nombre de couches

À présent, l'effet de l'ajout de couches de contraintes sur la qualité de la représentation est évalué. Pour ce faire, nous avons d'abord fait des optimisations avec la couche de base seulement, puis activé et désactivé les couches afin d'évaluer les résultats avec différentes combinaisons de couches actives. Les résultats sont portés en graphique sur la figure 5.7. Notons qu'il y a davantage de résultats avec une ou deux couches actives, simplement parce que plus de combinaisons sont possibles. Si seule la couche de base est présente, on n'indique qu'aucune couche n'est activée par un "0". Pour une couche (1), il y a successivement la couche numéro un, deux et trois qui est activée. Pour deux couches (2), les combinaisons suivantes ont été essayées : 12, 13, 14 et 23. Pour trois couches (3), les combinaisons 123, 124 et 134 ont été testées. Enfin, pour quatre couches (4), elles sont forcément toutes activées et un seul point par profil est obtenu sur la figure 5.7.

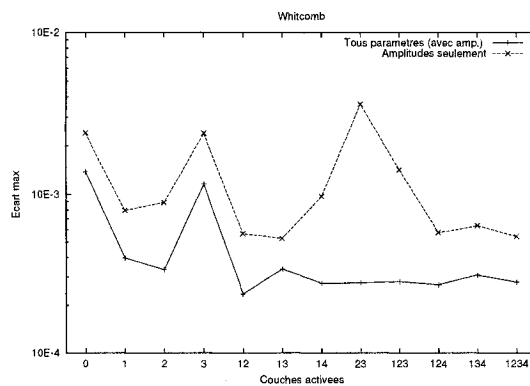
La figure 5.8 montre en détail, par profil et par combinaison de couches activées, les écarts maximum obtenus. Nous remarquons sur cette dernière figure, qu'à l'exception des cas où seule la couche de base est présente, que généralement les écarts maximum sont obtenus lorsqu'il n'y a que la couche trois d'activée. C'est que les optimisations avec celle-ci sont plus difficile à conduire, car les deux points clés que contient cette



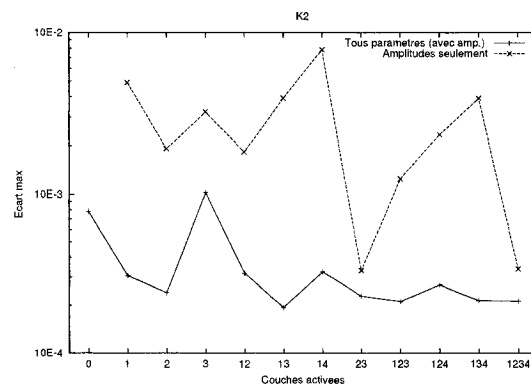
(a) RAE2822



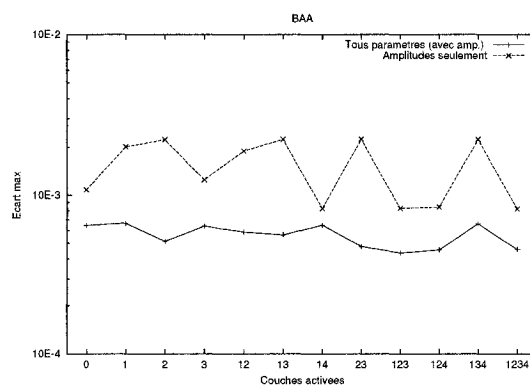
(b) bacj



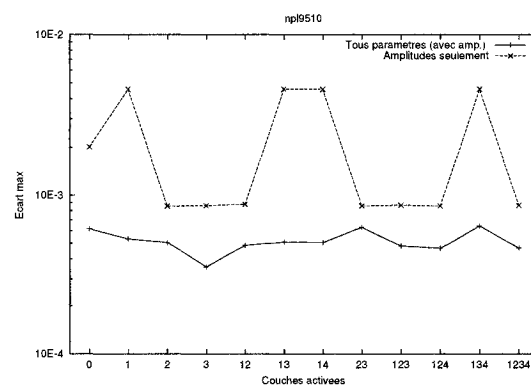
(c) Whitcomb



(d) k2



(e) BAA



(f) npl9510

FIG. 5.6: Comparaison entre les optimisations où seules les amplitudes sont optimisées et celles où tous les paramètres le sont pour les différents profils (trait plein : optimisation avec amplitudes, tireté : optimisation avec tous les paramètres)

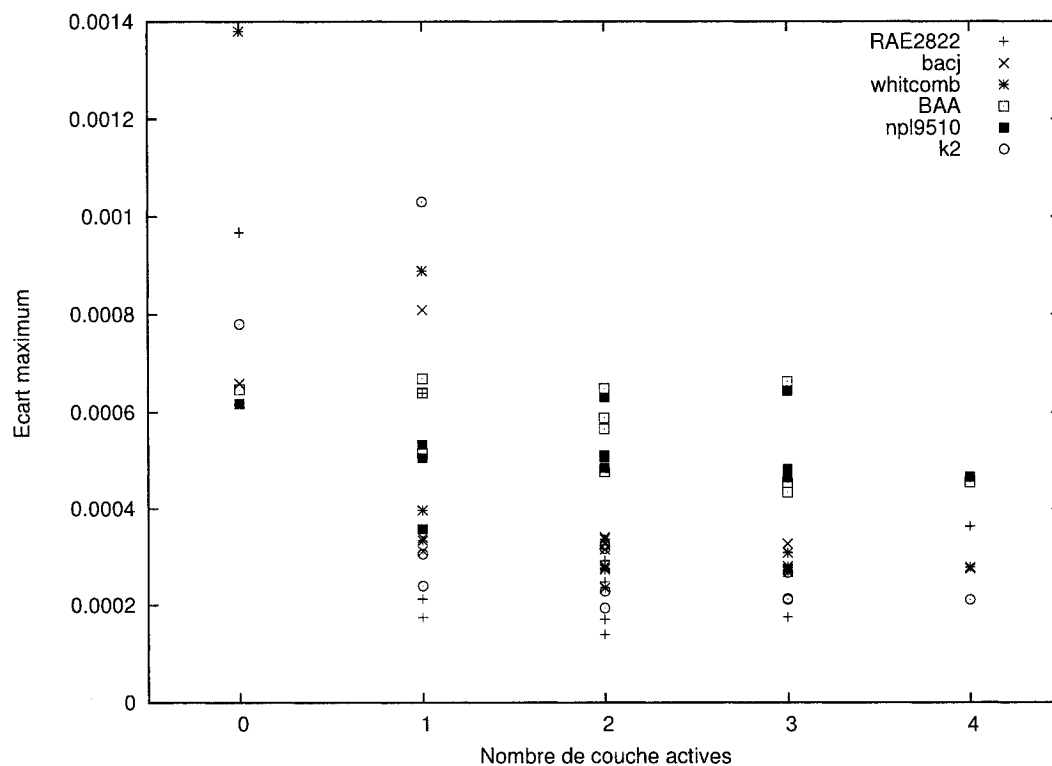
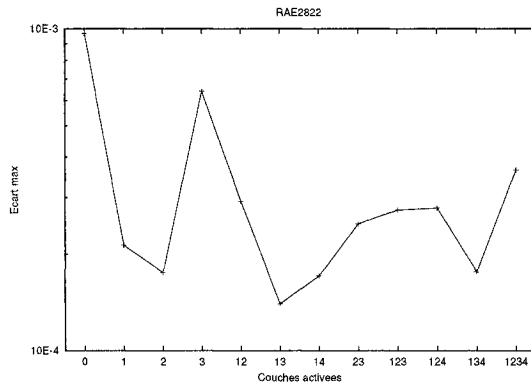


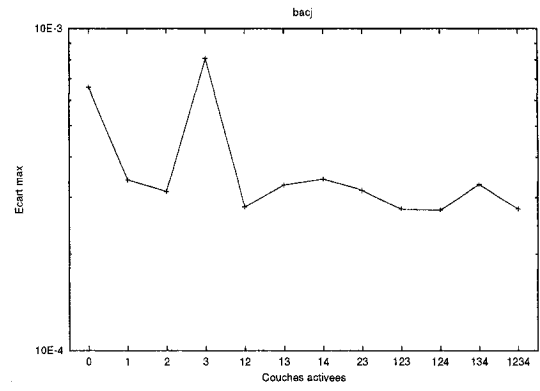
FIG. 5.7: Écart maximum en fonction du nombre de couches, tous profils confondus

couche, ont leur mouvement selon l'axe x libre, contrairement aux autres points clés. Il en résulte qu'un plus grand nombre de variables de design contrôlent ces points clés. De plus, lorsqu'il n'y a que cette couche d'activée, ces deux points clés sont proches l'un de l'autre. Il en résulte un couplage plus fort entre les paramètres et l'erreur de représentation, ce qui nuit aux optimisations.

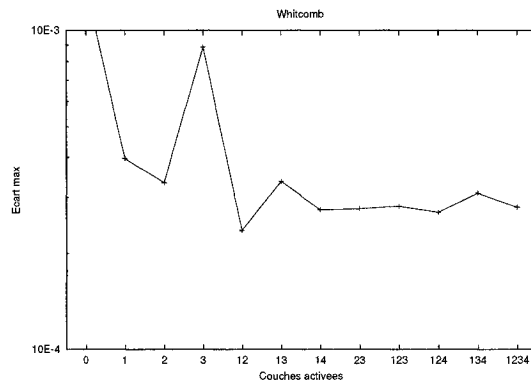
On doit s'attendre à ce qu'au fur et à mesure que des couches sont activées, l'erreur de représentation diminue. Cette tendance générale est observée sur la figure 5.7, mais elle n'est pas absolue. En effet, nous notons qu'il y a certains points qui dérogent à cette tendance, soit en présentant de meilleurs résultats avec un faible nombre de couches activées, soit en se détériorant en activant de nouvelles couches. Il y a plusieurs raisons pouvant expliquer la nature de ce problème, dont un fort couplage entre les paramètres géométriques et la fonction coût, le design initial trop loin de l'optimum, la nature de la



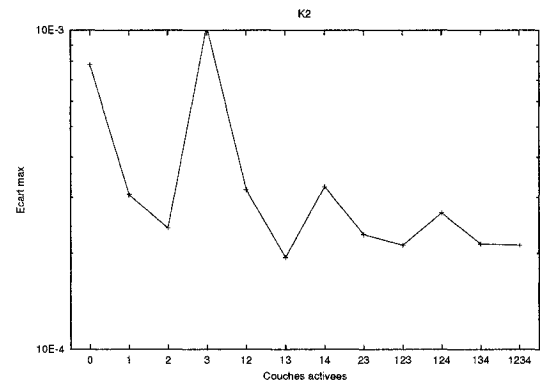
(a) RAE2822



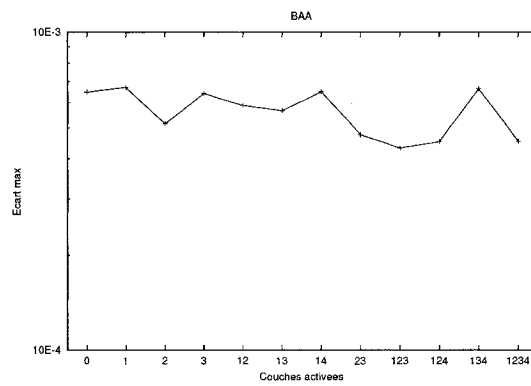
(b) bacj



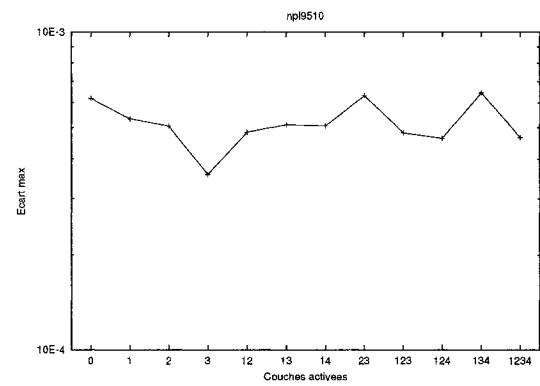
(c) Whitcomb



(d) k2



(e) BAA



(f) np19510

FIG. 5.8: Écart maximum en fonction des couches activées et par profil

fonction coût qui serait mal adaptée au problème d'optimisation, le trop grand nombre de variables de design ou le pas de différenciation choisi.

Afin de contourner le fort couplage nuisant à l'obtention de meilleurs résultats avec toutes couches activées, les caractéristiques géométriques des meilleures optimisations à une, deux et trois couches sont utilisées afin de générer un second design initial pour l'optimisation avec toutes couches activées et les résultats sont représentés sur la figure 5.9. Cette figure illustre à la fois la fonction coût et l'écart maximum. Nous voyons que la représentation est toujours améliorée en utilisant ce nouveau point de départ, mais parfois de très peu. Nous constatons aussi que pour le profil *bacj*, l'écart maximum n'est à peu près pas changé, mais que la fonction coût est diminuée, ce qui s'explique par le fait que l'erreur moyenne, donc la représentation en générale, est améliorée. À la lumière de cette expérience, nous pouvons affirmer que le point de départ a une importance cruciale pour l'optimisation et est donc déterminant sur la qualité de la représentation obtenue.

Étant donnée cette remarque, il serait intéressant d'essayer de conduire une optimisation à l'aide de la technique du recuit simulé. En effet cette technique relaxe les contraintes pour se sortir d'un minimum local afin de minimiser davantage la fonction coût. Il y a donc des similitudes avec la stratégie que nous avons utilisé pour trouver les nouveaux points de départ.

En fonction du nombre de variables de design, le nombre d'itérations requis pour trouver un minimum varie d'une vingtaine à une centaine. Le nombre d'évaluations de la fonction coût se situe quant à lui autour de deux à cinq mille. Notons cependant que certaines optimisations ont nécessité près de deux cents itérations pour environ quinze milles évaluations de la fonction coût.

Pour conclure cette section, nous présentons quelques résultats issus de ces optimisations. Nous voyons sur les figures 5.10 les six profils cibles superposés à ceux générés

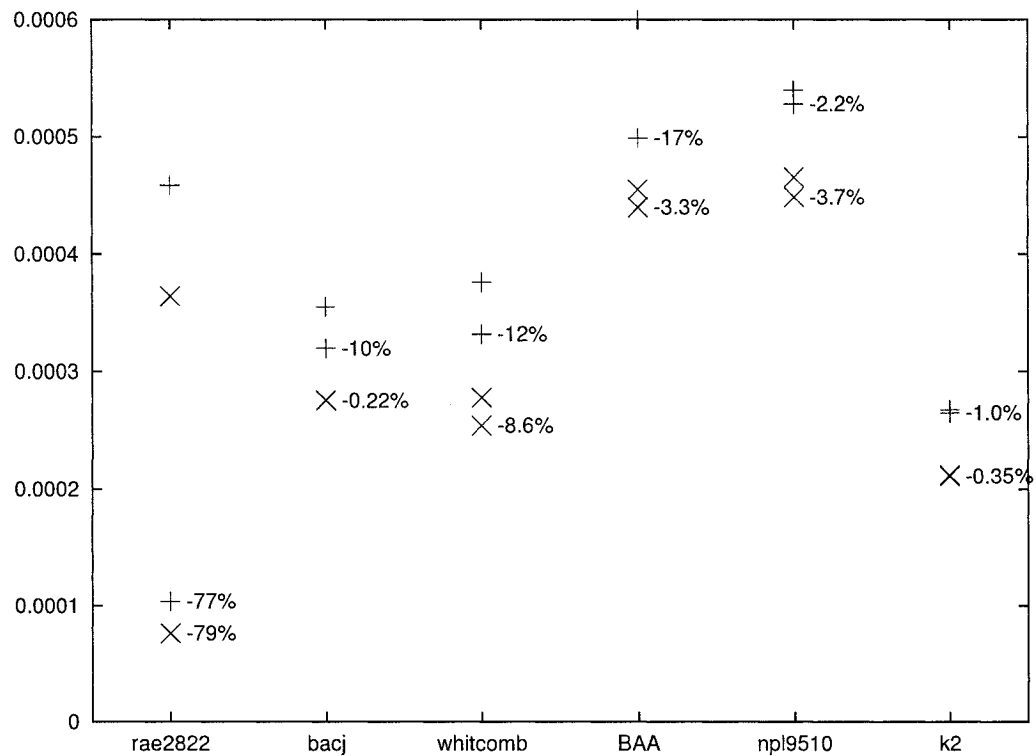


FIG. 5.9: Comparaison des écarts maximums (X) et des fonctions coûts (+) avec toutes les couches activées en fonction du design initial (point de départ) et pourcentage d'amélioration (%)

par la paramétrisation. Sous les différentes figures, on donne le nom du profil, suivi d'une suite de nombres mentionnant quelles couches sont actives et enfin la valeur de la fonction coût correspondante. Il est intéressant de constater que les plus grandes erreurs de représentation se situent généralement près du bord d'attaque, là où les variations de courbure sont très fortes. Ceci est observé visuellement quand la ligne du profil paramétré n'est pas exactement superposée à la ligne pointillée du profil cible. Malgré ces faibles erreurs de représentation, nous pouvons affirmer que la méthode de paramétrisation permet de représenter avec précision des profils existants.

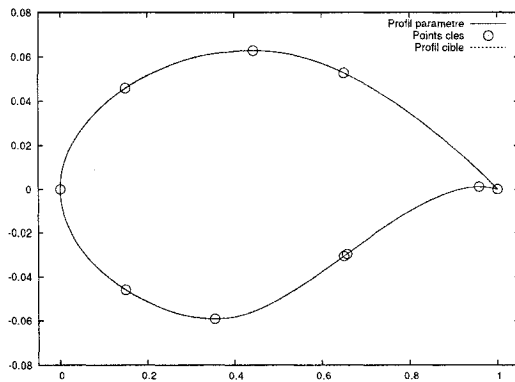
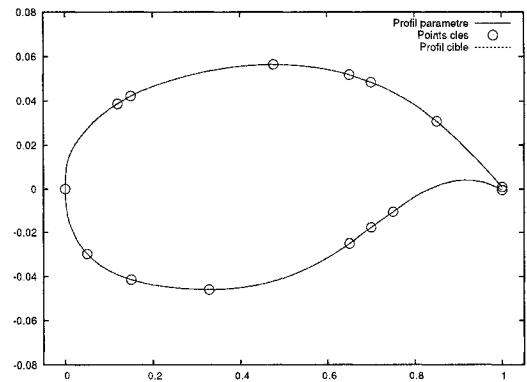
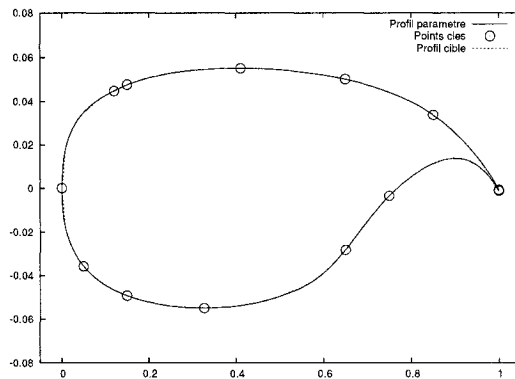
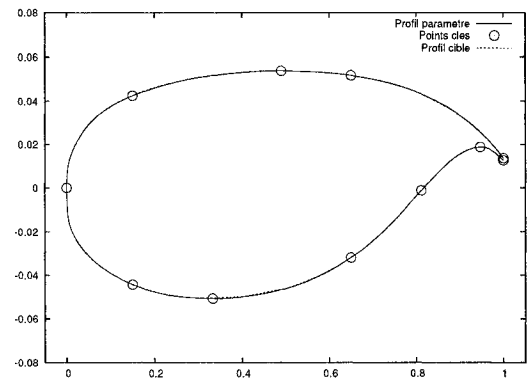
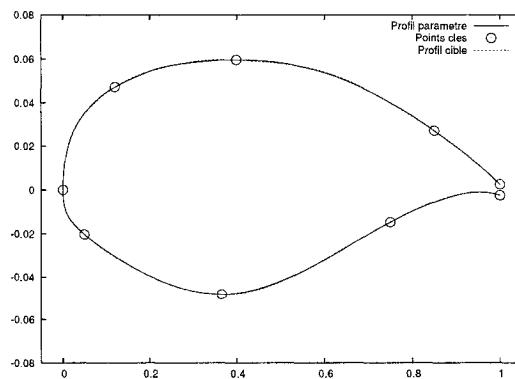
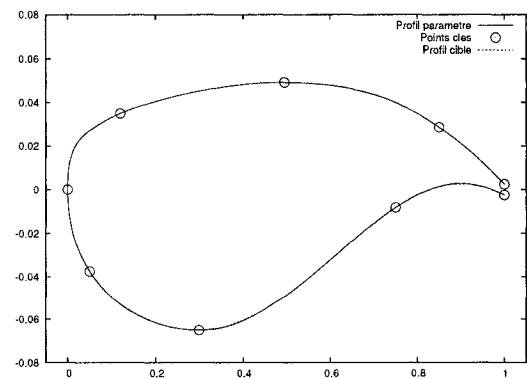
(a) RAE2822 ; 1 et 3 ; $2.32e-4$ (b) bacj ; 1, 2 et 4 ; $3.33e-4$ (c) Whitcomb ; 1 et 2 ; $3.21e-4$ (d) k2 ; 1 et 3 ; $2.98e-4$ (e) BAA ; 2 ; $5.46e-4$ (f) npl9510 ; 2 ; $2.64e-4$

FIG. 5.10: Résultats d'optimisation géométrique de différents profils (Nom du profil ; numéro des couches actives ; valeur de la fonction coût correspondante)

5.3 Validation géométrique sur l'aile typique BAW

Dans cette section, nous validerons la paramétrisation avec une aile transsonique typique qui nous a été fournie par Bombardier Aéronautique (*BAW - Bombardier Aeronautic Wing*). Pour cette validation, l'idée générale demeure la même, à savoir que nous optimisons un vecteur de design représentant un sous-ensemble des paramètres nécessaires à la génération de l'aile à l'aide de la méthode quasi-Newton BFGS.

5.3.1 Mise en place du problème d'optimisation tridimensionnel

Ici encore, la fonction coût est une mesure de l'erreur entre la géométrie cible et l'aile paramétrée. L'aile cible est définie par une série de listes de points correspondant au profil des différentes stations, car c'est l'approche utilisée par BA. Ces stations sont situées à l'emplanture, à mi-chemin de la partie interne, à la cassure, au tiers et au deux tiers de la partie externe et au bout. Comme la géométrie donnée en sortie de l'exécution du programme est également fournie sous le même format, la mesure de l'erreur est facile à implémenter.

Le point de départ (design initial) au problème d'optimisation géométrique de l'aile, qui est donné aux tableaux 5.2 et 5.3, a été extrait des caractéristiques géométriques de l'aile cible, tant pour la forme en plan que pour les points clés des stations guides. Cependant, comme l'aile typique BAW ne comporte pas de longerons ni de surfaces hypersustentatrices, il a fallu décider arbitrairement de l'emplacement des longerons et des bec et volet pour pleinement définir le modèle. Ici encore, des valeurs typiques, proposées par les ingénieurs de BA, ont été utilisées et sont données au tableau 5.2. Les caractéristiques géométriques ont été extraites aux endroits correspondants sur les différents profils.

TAB. 5.2: Définition complète de la forme en plan de l'aile BAW

Paramètres	Description
140.9100	Corde à l'emplanture
94.8780	Corde à la cassure
49.8910	Corde au bout
323.4730	Demi-envergure
47.6100	Station de l'emplanture
148.0193	Station de la cassure
27.3266	Angle de flèche
15	%C du longeron avant à la cassure
60	%C du longeron arrière à la cassure
25	%C du longeron avant au bout
55	%C du longeron arrière au bout
10	%C du point de recouvrement SE à la cassure
16	%C du point de recouvrement SE au bout
5	%C du point de recouvrement SI à la cassure
4.5	%C du point de recouvrement SI au bout
80	%C du point de recouvrement FE à la cassure
80	%C du point de recouvrement FE au bout
80	%C du point de recouvrement FI à la cassure
80	%C du point de recouvrement FI au bout
#Twist	
3.32297 0.84068 -1.31812	— Torsion de l'aile au R B T en degré
#Élévation	
2.079970 0.351880 -0.289370	— Élévation du bord d'attaque au R B T
#Translation en X	
261.8180	

Le nombre de variables de design varie de 48 à 172, en fonction du nombre de couches activées. Donc, un nombre plus élevé d'évaluations de la fonction coût est nécessaire afin de faire le calcul des gradients, comparativement au cas 2D. Cette opération devient alors plus longue, d'autant plus que la génération d'une aile ainsi que l'évaluation de la fonction coût prennent aussi davantage de temps. Il n'est donc pas rare d'atteindre des nombres de dix ou vingt mille, voir davantage, comme nombre d'évaluations de la fonction coût. De plus, le nombre d'itérations est plus élevé et dépasse régulièrement la centaine, car les améliorations de la fonction coût sont faibles à chaque itération. Il

Tab. 5.3: Point de départ et bornes des optimisations pour les caractéristiques géométriques aux profils guides de l'aile BAW
(Les amplitudes ne sont pas présentes, car elles sont toutes 1 au départ et les bornes sont [0.1, 2] partout)

Couche	Nom PC	Position X	Position Z	Angle	Courbure
0	Valeurs Bornes	TE _{in}			
		Provient de la forme en plan	-0.0025	1.057	-1.045
		Pas optimisé			
	Valeurs Bornes	IN	0.3606		-1.257
		Provient de la forme en plan	0.3677		
		Pas optimisé			
	Valeurs Bornes	LE			
		Provient de la forme en plan	[0.3,0.5]		
		Pas optimisé			
	Valeurs Bornes	EX	0.4112		
		Provient de la forme en plan	[0.3,0.5]		
		Pas optimisé			
1	Valeurs Bornes	RSI			
		Provient de la forme en plan	-0.0606		
		Pas optimisé			
	Valeurs Bornes	RST			
		Provient de la forme en plan	-0.0422		
		Pas optimisé			
	Valeurs Bornes	RSE			
		Provient de la forme en plan	[0.06,0.08]		
		Pas optimisé			
	Valeurs Bornes	RSE			
		Provient de la forme en plan	[0.05,0.07]		
		Pas optimisé			
2	Valeurs Bornes	FI			
		Provient de la forme en plan	-0.0125		
		Pas optimisé			
	Valeurs Bornes	SI			
		Provient de la forme en plan	-0.0289		
		Pas optimisé			
	Valeurs Bornes	SE			
		Provient de la forme en plan	[0.05,0.07]		
		Pas optimisé			
	Valeurs Bornes	FE			
		Provient de la forme en plan	[0.01,0.04]		
		Pas optimisé			
3	Valeurs Bornes	Cavité			
		Provient de la forme en plan	0.9532		
		Pas optimisé			
	Valeurs Bornes	Inflexion			
		Provient de la forme en plan	[0.65,0.75]		
		Pas optimisé			
4	Valeurs Bornes	SSI			
		Provient de la forme en plan	-0.0307		
		Pas optimisé			
	Valeurs Bornes	SSI			
		Provient de la forme en plan	-0.0344		
		Pas optimisé			

en résulte donc que le temps des optimisations augmente grandement et plusieurs heures sont requises pour mener à terme de bonnes optimisations. Pour ces raisons, moins d'optimisations ont été menées et en conséquence aucune étude sur le pas de différenciation à prendre n'a été conduite. Il a été choisi à 10^{-4} étant donné que c'est celui qui a donné les meilleurs résultats dans le cas 2D.

5.3.2 Effet de la fonction coût

Comme l'aile est une entité dimensionnelle, le choix de la fonction coût est plus difficile à faire. En effet, nous disposons d'autant de fonctions coûts "2D" que nous avons de stations de définition. Sur chacune de ces stations, les distances maximales et moyennes sont obtenues comme précédemment. Mais si la somme de ces erreurs de représentation est utilisée pour constituer la fonction coût du problème d'optimisation de l'aile, les stations près de l'emplanture, qui ont des plus grandes cordes, donc possiblement des erreurs également plus grandes, risquent de camoufler la contribution de celles près du bout, menant à des optimisations qui favoriseraient les régions près de l'emplanture. D'un autre côté, les distances constituant la mesure de l'erreur peuvent être divisées par la corde de la station courante afin d'obtenir une mesure relative. La fonction coût devient alors la somme de ces erreurs et chaque station a la même contribution dans la fonction coût et donc, l'optimiseur tente d'améliorer la représentation de toutes les stations également. Cependant, comme les erreurs sont équivalentes d'une station à l'autre, les distances entre les géométries paramétrées et cibles risquent d'être plus grandes à l'emplanture qu'au bout. Ces deux approches sont comparées pour déterminer laquelle semble la mieux adaptée à l'optimisation géométrique.

Les erreurs de représentations des stations 2D se calculent avec la même stratégie qu'à la section précédente. La première fonction coût, que nous nommons la fonction coût dimensionnelle, s'exprime comme suit :

$$f(\mathbf{x}) = 2 \left(c_{moy} \varepsilon_{moy_{rel}}(\mathbf{x}) \right) + \varepsilon_{max_{abs}}(\mathbf{x}) \quad (5.5)$$

où c_{moy} est la corde moyenne de l'aile, $\varepsilon_{moy_{rel}}(\mathbf{x})$ est la moyenne des erreurs moyennes relatives aux stations de définition et $\varepsilon_{max_{abs}}(\mathbf{x})$ est l'erreur maximale absolue sur l'aile entière. La seconde fonction coût, nommée fonction coût adimensionnelle, s'exprime comme suit :

$$f(\mathbf{x}) = 2\varepsilon_{moy_{rel}}(\mathbf{x}) + \varepsilon_{max_{rel}}(\mathbf{x}) \quad (5.6)$$

où $\varepsilon_{max_{rel}}(\mathbf{x})$ est l'erreur maximale relative. La moyenne des erreurs moyennes relatives et les erreurs maximales relatives et absolues se calculent comme suit :

$$\begin{aligned} \varepsilon_{moy_{rel}} &= \frac{1}{k} \sum_{j=1}^k \epsilon_{moy_j}(\mathbf{x}) \\ \varepsilon_{max_{rel}} &= \max |\bar{\epsilon}_{max_j}| \\ \varepsilon_{max_{abs}} &= \max |\epsilon_{max_j}| \end{aligned} \quad (5.7)$$

où k représente le nombre de stations et

$$\begin{aligned} \epsilon_{moy_j} &= \frac{1}{nc_j} \sum_{i=1}^n \epsilon_i(\mathbf{x}) \\ \bar{\epsilon}_{max_j} &= \frac{\max |\epsilon_i|}{c_j} \\ \epsilon_{max_j} &= \max |\epsilon_i| \end{aligned} \quad (5.8)$$

où c_j est la corde de la $j^{ième}$ station et ϵ_i représente une des 500 mesures de la distance.

Ces deux fonctions coût ont été essayées sur des séries d'optimisations et les résultats sont comparés. Pour y parvenir, les fonctions coûts ont été calculées sur chaque design

final obtenu (design *a* et *d*). Ainsi, sur les résultats de la série adimensionnelle (design *a*), la fonction coût dimensionnelle a été calculé et vice-versa. Le tableau 5.4 résume les résultats obtenus. Nous constatons, comme attendu, que les optimisations menées avec une fonction coût particulière donnent de meilleures valeurs de celles-ci que si les optimisations ont été menées avec l'autre fonction coût (valeur avec astérisque dans le tableau). Ces valeurs serviront de références pour la comparaison illustrée à la figure 5.11. Sur celle-ci, les détériorations représentant le pourcentage de l'écart entre la référence et l'évaluation avec la fonction coût n'ayant pas servi à son obtention, sont portées en graphique. Nous y constatons que le design *a* garde une meilleure qualité lorsque évalué avec l'autre fonction coût, avec une détérioration d'un peu plus de 14%. Ceci suppose que les optimisations menées avec la fonction adimensionnelle donnent de meilleurs designs, car elles approchent mieux l'aile cible peu importe la façon de mesurer l'erreur.

TAB. 5.4: Résultats obtenus lors des optimisations géométriques de l'aile BAW à l'aide des deux différentes fonctions coûts

Valeur finale de la fonction coût ↓	Optimisation avec la fonction coût		Écart (%)
	adimensionnelle (design <i>a</i>)	dimensionnelle (design <i>d</i>)	
Adimensionnelle :	8.19722E-03*	9.68316E-03	18.1
Dimensionnelle :	1.01094	0.884046*	14.4

Le détail de l'évaluation de la distance entre l'aile cible et les designs obtenus (valeur avec astérisque dans le tableau 5.4) est donné au tableau 5.5. On y présente, pour les deux fonctions coût, la corde des six stations de l'aile avec les distance et erreurs moyennes et maximales. À l'aide de ce tableau, nous constatons que la fonction coût adimensionnelle tend à égaliser les erreurs maximum relatives, ce qui est aussi visible par les distances maximum absolues qui sont plus petites au fur et à mesure que la corde des profils diminue. De son côté, la fonction coût dimensionnelle à tendance à uniformiser les erreurs maximum absolues, donc les erreurs relatives augmentent plus la corde est petite. Ces effets sont ceux attendus par la nature des fonctions coût du problème d'optimisation.

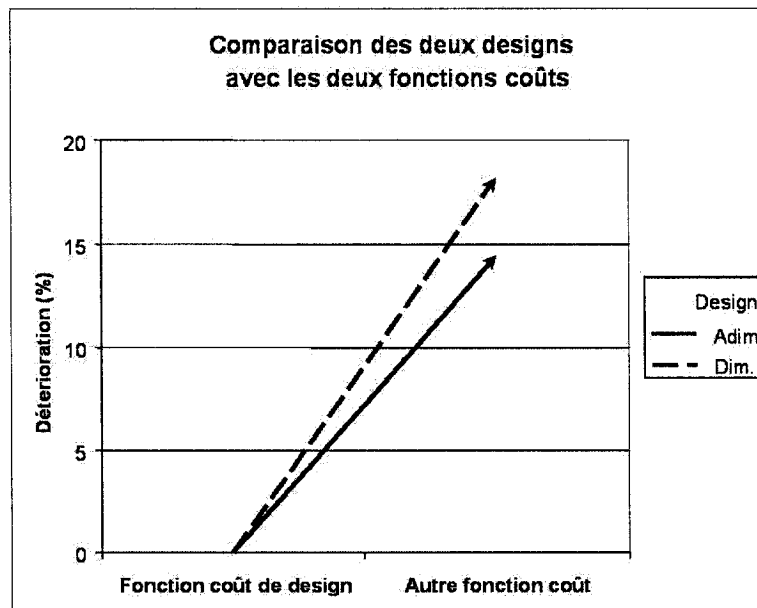


FIG. 5.11: Représentation de la détérioration d'un design lorsque évalué à l'aide de l'autre fonction coût (fct. dim. sur design *a* et fct. adim. sur design *d*)

5.3.3 Effet du nombre de couches

À présent, nous voulons constater l'effet de l'ajout de couches sur la qualité de la représentation. Pour ce faire, plusieurs optimisations ont été menées avec différentes caractéristiques, mais toujours avec le même point de départ donné aux tableaux 5.3 et 5.2 afin de n'illustrer que l'effet de l'ajout de couches. Les résultats pour les deux fonctions coûts sont présentés à la figure 5.12.

Cette figure montre que la diminution attendue de l'erreur de représentation au fur et à mesure que les couches sont activées n'est pas présente. De plus, selon la fonction coût utilisée, l'effet sur la qualité de la représentation n'est pas le même. En effet, avec la fonction coût adimensionnelle, les meilleures représentations sont trouvées lorsqu'il n'y a qu'une couche activée, et l'ajout de nouvelles couches ne fait guère varier. La meilleure représentation est obtenue avec la couche 2 active seulement (figure 5.13). Pour la fonction coût dimensionnelle, aucune tendance n'est observée avec l'ajout de

TAB. 5.5: Détails des distances et des erreurs sur les stations de l'aile après les optimisations

		Opt. avec fct. coût adim.	Opt. avec fct. coût dim.		
		Distance (erreur) absolue			
Station	Corde	Moyenne	Maximum	Moyenne	Maximum
1	141.15	0.0776	0.748	0.0803	0.618
2	117.99	0.195	0.626	0.143	0.624
3	94.89	0.158	0.450	0.182	0.623
4	79.89	0.0907	0.258	0.111	0.490
5	64.89	0.173	0.344	0.189	0.596
6	49.90	0.0502	0.198	0.0271	0.196

		Distance (erreur) relative			
Station	Corde	Moyenne	Maximum	Moyenne	Maximum
1	141.15	5.50E-4	5.30E-3	5.69E-4	4.38E-3
2	117.99	1.65E-3	5.30E-3	1.22E-3	5.28E-3
3	94.89	1.67E-3	4.74E-3	1.91E-3	6.57E-3
4	79.89	1.14E-3	3.23E-3	1.40E-3	6.14E-3
5	64.89	2.67E-3	5.30E-3	2.91E-3	9.19E-3
6	49.90	1.01E-3	3.97E-3	5.44E-4	3.94E-3

couches de contraintes. Lorsque 2 couches sont activées, les valeurs de fonction sont distribuées très irrégulièrement. Dans ce cas, la meilleure représentation est obtenue avec les couches 1 et 2 actives (figure 5.14). La section suivante explique les raisons pour lesquelles la tendance d'amélioration avec l'ajout de couche n'est pas rencontrée comme escompté.

Les deux meilleures ailes de chaque série d'optimisation sont présentées aux figure 5.13 et 5.14. Sur celles-ci sont présentées la distribution d'erreur (à gauche) et de distance (à droite) sur les ailes obtenues. Nous constatons encore, comme les échelle employées sont les mêmes pour les deux figures, que la fonction coût adimensionnelle minimise les erreurs, alors que l'autre minimise les distances. Ceci est observé par la présence ou non du rouge (distance et erreur maximales) sur les profils cibles.

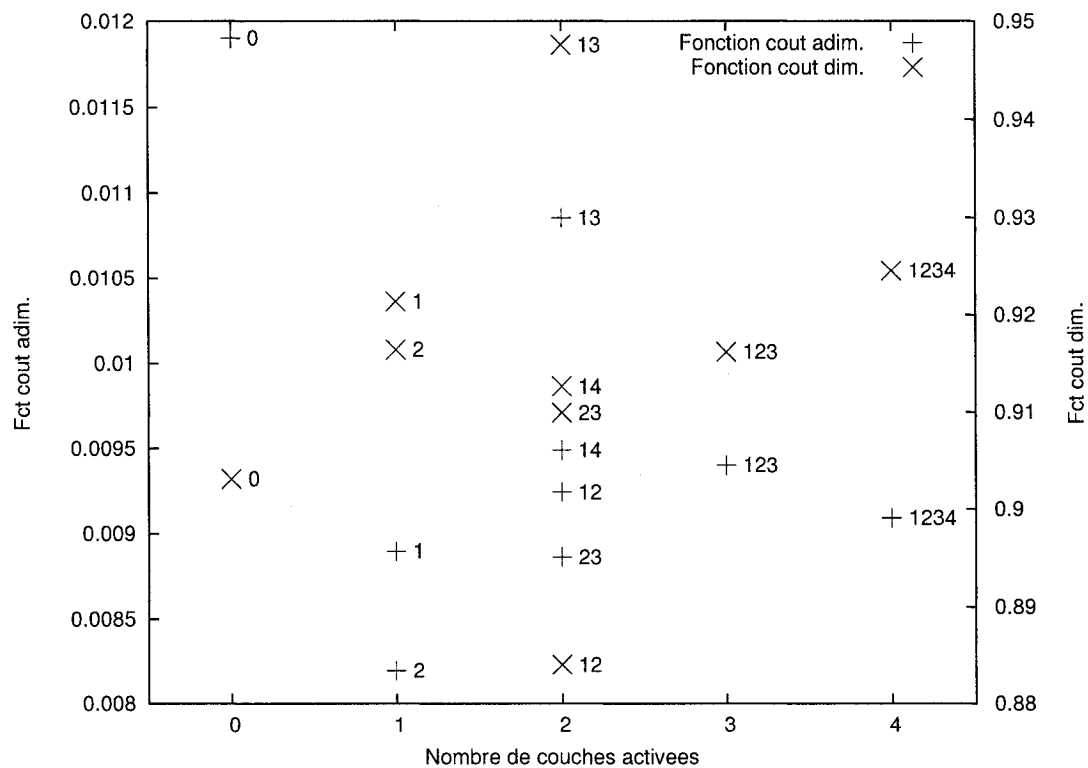


FIG. 5.12: Fonctions coût dimensionnelle (échelle à droite) et adimensionnelle (échelle à gauche) en fonction des couches actives pour l'aile BAW

5.3.4 Comparaison des variations des caractéristiques géométriques de l'aile et des coniques servant à l'interpolation dans le sens de l'envergure

On a déjà noté que l'approximation de l'aile ne colle pas exactement sur les stations de références et que cette approximation ne s'améliore pas nécessairement quand de nouvelles couches sont activées. Comme il a été prouvé que la paramétrisation 2D peut représenter des profils avec un faible niveau d'erreur, les problèmes de précision au niveau de l'aile proviennent des interpolants coniques utilisés dans le sens de l'envergure.

Les interpolants de la position de ces caractéristiques géométriques sur toutes l'envergure ainsi que les stations paramétrées et celles de définition sont représentées sur la figure 5.15. Cette figure nous aide à comprendre pourquoi l'ajout de nouvelles couches

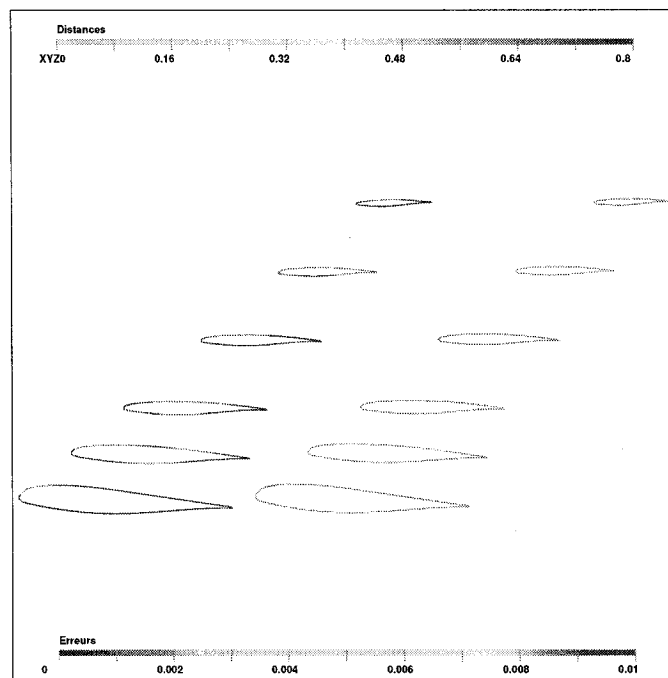


FIG. 5.13: Erreur de représentation pour l'aile obtenue avec la fonction coût adimensionnelle et la couche 2 activée

n'aboutit pas nécessairement à la diminution des erreurs de représentation. La nature des interpolants ne convient pas toujours pour représenter la variation des paramètres de l'aile cible correctement. Nous illustrons ceci en isolant chacun des paramètres dont la variation selon l'envergure est paramétrable. Pour ce faire, les paramètres sélectionnés sont extraits des différentes stations de l'aile cible afin d'avoir leur évolution selon l'envergure. Cette dernière est comparée à l'interpolant conique de la section 3.2 généré automatiquement.

Premièrement, comme nous l'avons mentionné plus haut, l'aile cible ne contient pas explicitement des surfaces hypersustentatrices. Il a fallu décider de leurs emplacements à l'aide de valeurs typiques (voir tableau 5.2). Or, à ces positions sur l'aile cible, les variations selon l'envergure ne sont pas linéaires, comme on peut le voir sur la figure 5.15. Dès lors, il est impossible d'interpoler exactement les régions correspondantes aux bec et volet des stations cibles dans le sens de l'envergure, un compromis doit être fait

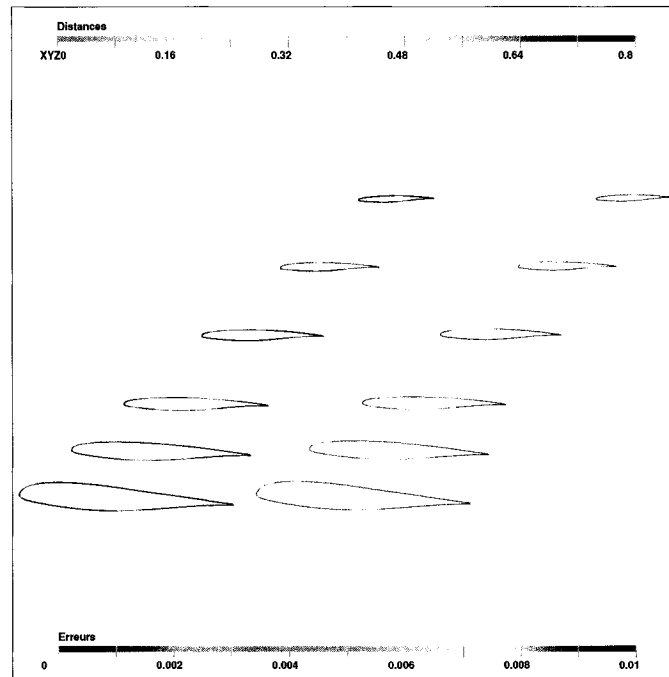


Fig. 5.14: Erreur de représentation pour l'aile obtenue avec la fonction coût dimensionnelle et les couches 1 et 2 activées

ce qui génère forcément des erreurs.

Deuxièmement, les variations des paramètres de l'aile cible dans le sens de l'envergure ne sont pas nécessairement coniques. Les longerons avant et arrière, eux aussi positionnés à l'aide de valeurs typiques, n'ont pas leur hauteur z qui varie suivant une conique, comme le montre les figures 5.16 a), b), c) et d). À l'aide de celles-ci, nous constatons que les erreurs entre les variations cibles des paramètres et les coniques générées automatiquement sont plus grandes sur l'extrados que sur l'intrados. De plus, toujours pour l'extrados des longerons, la courbe cible semble se rapprocher d'une conique dans la partie interne, mais d'une droite dans la partie externe, ce qui explique pourquoi les erreurs sont plus grandes à ces endroits. Pour l'intrados des longerons, les variations sont moins importantes et semblent se rapprocher de deux segments de droite, ce qui explique que dans ce cas, les erreurs semblent se trouver davantage dans la partie interne. Il est à noter que la projection de ces quatre courbes dans le plan xy donne une droite.

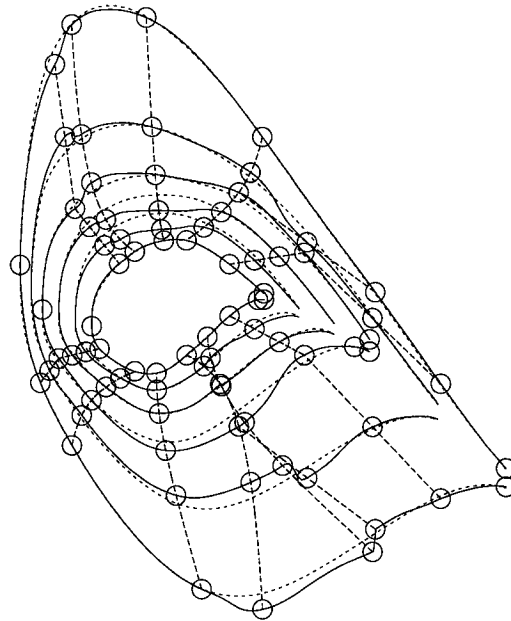
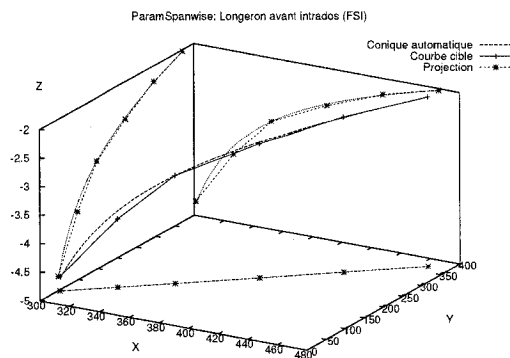
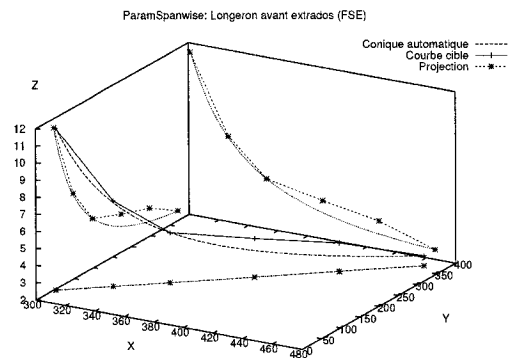


FIG. 5.15: Exemple d'aile à rejetée obtenue (trait plein) comparée à l'aile cible BAW (pointillé), représentation des différents interpolants coniques dans le sens de l'envergure (tireté) et points clés correspondant (O)

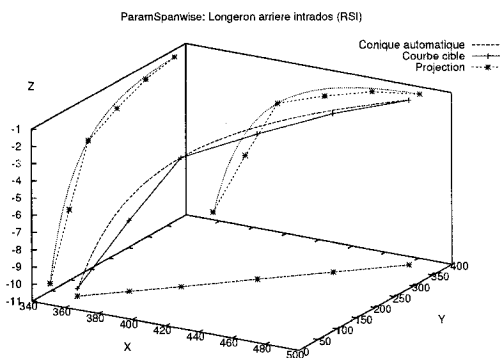
De plus, les points correspondant aux endroits les plus éloignés de la corde locale (points minimum et maximum), tout comme le point d'inflexion, ne correspondent pas à des coniques, comme le montre les figures 5.17. Pour les points maximum et minimum, qui correspondent à l'extrados et l'intrados, les mêmes observations que pour les longerons peuvent être faites, à savoir que sur l'extrados la courbe cible semble être une conique et un segment de droite dans la partie interne et externe respectivement et deux segments de droite pour l'intrados. De plus, leurs projections dans le plan xy se rapprochent aussi d'une droite. C'est au point d'inflexion que les erreurs sont les plus grandes, car aucune projection ne se réduit à une droite. Les caractéristiques géométriques de ce point clé de l'aile cible varient encore selon deux segments de droite.



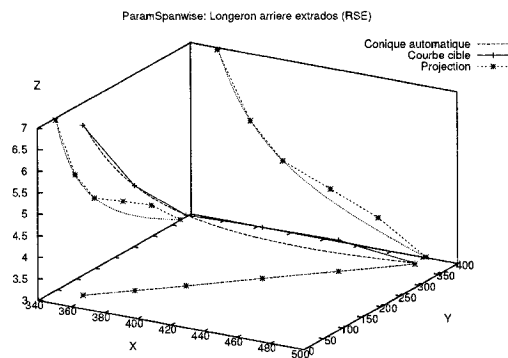
(a) Variation du longeron avant à l'intrados



(b) Variation du longeron avant à l'extrados

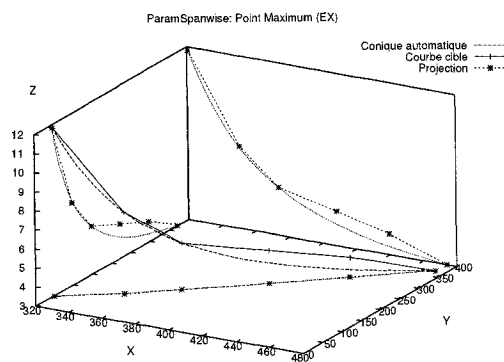


(c) Variation du longeron arrière à l'intrados

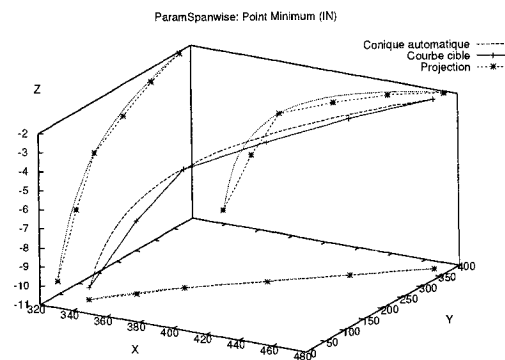


(d) Variation du longeron arrière à l'extrados

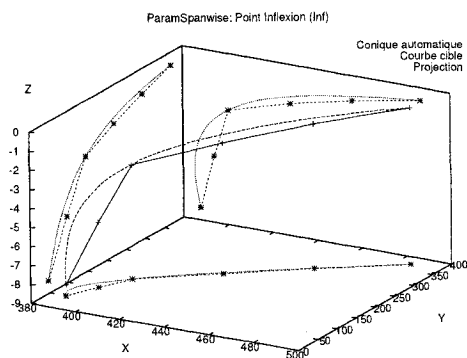
FIG. 5.16: Variation dans le sens de l'envergure de la position des longerons sur l'aile cible et coniques résultantes



(a) Variation de la position du point maximum



(b) Variation de la position du point minimum



(c) Variation de la position du point d'inflexion

FIG. 5.17: Variation dans le sens de l'evergure des paramètres du point maximum, minimum et d'inflexion sur l'aile cible et coniques résultantes

Troisièmement, la cavité est dans le volet de l'aile. Cette caractéristique n'est donc gouvernée qu'avec les trois profils guides puisque cette surface est réglée. De plus, le profil à l'implanture de l'aile cible ne comporte pas de cavité, ce qui se trouve à être un cas particulier plus difficile à gérer par le programme.

Quatrièmement, le longeron auxiliaire, qui est un segment de droite, prend origine à la position du longeron arrière à la cassure et se prolonge parallèlement au bord de fuite jusqu'à l'implanture. Peu de contrôle est donc possible sur cette entité structurelle.

Aussi, il est à noter que le point d'inflexion et le longeron auxiliaire se croisent. Ce croisement crée des oscillations sur les profils générés (figure 5.15), car aucune contrainte ne relie ces deux interpolants. Enfin, la proximité entre le longeron avant, qui est défini à l'aide d'une conique, et le bec défini linéairement cause des problèmes dus à la trop grande différence entre ces interpolants.

CONCLUSION

Au cours de ce travail, un survol des techniques de paramétrisation courantes et un classement sommaire de celles-ci ont été présentés. Nous avons par la suite élaboré les critères de sélection qui nous ont permis de sélectionner les deux types de courbes retenues.

Dans un deuxième temps, un court rappel sur les écoulements transsoniques est présenté, suivi d'une étude détaillée des besoins et contraintes afin de s'assurer que la paramétrisation développée sur ceux-ci répond bien à nos objectifs. Les cas 2D pour les profils et la forme en plan ont été exposés suivis de ceux pour l'aile entière.

Par la suite, nous avons développé plusieurs méthodes génériques d'interpolation en 2D. Deux de ces méthodes sont basées sur les B-Splines : le collage de B-Splines contraintes aux extrémités et les B-Splines multi-contraintes en plusieurs endroits (interpolation avancée de points clés). La troisième méthode se base sur les coniques et sert à l'interpolation des caractéristiques géométriques des points clés dans le sens de l'envergure.

Enfin, nous avons élaboré une paramétrisation afin de générer un squelette de contraintes géométriques afin d'obtenir une surface répondant aux contraintes des ailes transsoniques avec cassure. Deux implantations, l'une se basant sur les paramètres PARSEC et l'autres sur les B-Splines, ont été faites utilisant cette approche.

Finalement, nous avons développé une structure permettant l'optimisation géométrique de profil 2D et d'aile 3D. Dans chacun des cas, des fonctions coûts représentatives des erreurs de représentation ont été développées. Puis, plusieurs optimisations ont été conduites, ce qui nous a permis de commenter la paramétrisation tant pour le cas bi-dimensionnel que tridimensionnel, en plus de nous orienter sur les prochaines voies à suivre.

6.1 Contribution en fonction des objectifs énumérés

Les trois principaux objectifs du travail, énumérés en introduction, ont bien été rencontrés même si certains points restent encore à améliorer. Le premier objectif de la thèse, à savoir l'élaboration d'une nouvelle paramétrisation adaptée aux ailes transsoniques avec cassures et angle de flèche, a été solutionné à l'aide de deux classes d'interpolants, l'une selon la corde de l'aile et l'autre selon l'envergure. Au niveau bidimensionnel (corde), l'utilisation de points clés regroupés en couches significatives permet de générer des profils aérodynamiques adimensionnels valables à l'aide de B-Splines comme support mathématique. Au niveau de l'aile, l'emploi d'interpolants dimensionnels coniques selon l'envergure permet d'extraire les valeurs de ces points clés à toutes stations de l'aile, ce qui complète sa définition géométrique.

Ensuite, la création d'un programme informatique de génération de la surface d'une aile se basant sur cette paramétrisation a été réalisée avec succès. L'utilisation de bibliothèques de classes C++ développées à l'École Polytechnique de Montréal (*Pirate et opgeom*) a été grandement utile pour la manipulation de courbes et surfaces B-Splines. Ce programme peut écrire l'aile générée dans des fichiers ASCII sous plusieurs formats, selon ce que l'utilisateur désire. Ainsi, pour la visualisation, le format *Pirate* est utilisé, pour l'échange de données avec les tâches 1 et 5 du projet MOSAIC, des formats donnant la géométrie de l'aile à plusieurs stations à l'aide d'un nuage de points ont été développés.

Enfin, pour la validation des profils et ailes obtenus, une structure d'optimisation géométrique se servant de distances comme fonction coût et se basant sur la méthode quasi-Newton BFGS a été développée. À l'aide de celle-ci, au niveau bidimensionnel, plusieurs profils aérodynamiques courants ont pu être représentés à l'intérieur d'une certaine tolérance. Au niveau tridimensionnel, une aile transsonique typique a été représentée. À l'aide de ces expériences numériques, plusieurs voies d'améliorations nous sont appa-

rués, spécialement pour le cas de l'aile complète. Celles-ci seront discutées sommairement dans les prochaines pages.

6.2 Discussions

À la lumière des résultats présentés au chapitre 5, plusieurs améliorations peuvent être apportées à ce module et à la structure d'optimisation. Pour la validation bidimensionnelle (optimisation géométrique), nous avons relevé les lacunes suivantes :

1. Comme le nombre de variables de design augmente quand de nouvelles couches sont activées, celui-ci devient important et l'optimiseur à plus de difficultés à obtenir une bonne direction de descente et s'arrête dès qu'un minimum local est trouvé. Certaines variables doivent être fixées afin de les éliminer du vecteur de design de l'optimisation pour éviter cette situation. Le recours à un plan d'expérience pour déterminer quelles variables ont un grand impact sur la fonction coût et lesquelles en ont moins permettra de faire ce choix judicieusement.
2. Le point de départ étant très important pour les optimiseurs à gradient, celui-ci devra être choisi avec une plus grande attention, par exemple en extrayant les caractéristiques géométriques des optimisations précédentes au lieu de toujours partir du même point de départ.
3. De plus, il faut ajuster les bornes de chaque variable de design de façon plus adéquate afin de limiter la grandeur de l'espace de design pour éliminer par construction certains profils irréalisables. Il serait préférable de faire des ensembles de bornes adaptés pour chaque profil au lieu d'un ensemble commun.
4. Le pas de différenciation choisi a aussi une grande influence sur la qualité des optimisations. Le fait d'imposer le même pour toutes les variables nuit à l'obtention d'un bon minimum. Alors nous pourrions mener des études de sensibilité sur les

variables de design afin d'améliorer les représentations optimales obtenues. Grâce à celles-ci, le pas de différenciation optimale à prescrire pour chaque variable peut être trouvé et imposé.

5. De plus, la nature de la mesure de l'erreur, donc de la formulation de la fonction coût, peut également ne pas être totalement bien adaptée aux optimisations. Si celle-ci n'est pas bien corrélée avec la nature des paramètres gouvernant la génération des profils, il en résulte que les directions de descente sont difficiles à obtenir, donc les résultats en souffrent.
6. Finalement, avec toutes les couches activées, il y a plusieurs points clés qui se retrouvent très près les uns des autres, surtout vers l'arrière du profil et sur l'intrados, car les longerons arrière et auxiliaire ainsi que le point d'inflexion et les points associés au volet sont situés dans la même région. Il en résulte que plusieurs paramètres doivent évoluer de façon simultanée afin de réduire significativement la fonction coût.

Pour le cas de l'aile typique, en plus des points précédents, nous pouvons ajouter :

1. D'autres familles d'interpolant pour l'envergure doivent être développées. Les interpolants en deux segments de droite doivent être évités, car ils vont générer des discontinuités dans l'aile, correspondant à la station de la cassure, donc à la jonction des deux droites. Ce qui semble mieux adapté au problème de génération d'aile sont des coniques avec forte courbure à la cassure, qui s'apparente à deux droites tout en restant continue, ou des interpolants mixtes constitués du collage entre une droite et une courbe se rapprochant d'une conique.
2. Une méthode permettant d'éviter des oscillations de la géométrie en toutes circonstances doit être développée et implémentée. En effet, jusqu'ici, dans certains cas (combinaison des couches actives avec les valeurs de paramètres des points clés) les ailes générées peuvent présenter des bosses et des oscillations à leur sur-

face.

3. Développer une nouvelle fonction coût qui serait à mi-chemin entre les deux qui ont été proposées afin d'avoir une meilleure mesure de l'erreur de représentation.
4. Faire une validation géométrique de l'aile avec la deuxième méthode de construction des B-Splines (multi-contraintes - section 3.1.2) afin de l'améliorer, surtout en ce qui attrait à la construction du vecteur nodal, car cette étape est vraiment cruciale sur l'allure des courbe et donc sur la qualité des représentations obtenues.

6.3 Travaux futurs

Avec l'expérience accumulée au cours de ce travail, nous voyons que plusieurs avenues de perfectionnement se présentent, notons entre autres :

1. Ajouter une couche supplémentaire, dite libre, pouvant supporter n nouveaux points clés à des endroits quelconques pour un contrôle accru sur la surface à des emplacements qui ne correspondent pas nécessairement à des endroits où des composantes et des caractéristiques particulières sont présentes à la surface.
2. De plus, il serait possible d'améliorer la construction des interpolants coniques actuels en mettant en paramètre la pente et le rayon de courbure afin d'accroître le contrôle des variations des paramètres selon l'envergure. Une autre option serait de générer les coniques d'une façon non indépendante, *i.e.* que ces interpolants connaissent leurs positions relatives afin d'obtenir de meilleures variations des paramètres dans le sens de l'envergure.
3. Pour la méthode d'interpolation de collage de B-Splines contraintes aux extrémités, il serait prometteur d'inclure deux amplitudes par point clé, une pour la courbe de gauche et l'autre pour celle de droite. Nous avons noté que les plus grandes erreurs de représentation se trouvent habituellement près du bord d'attaque et donc,

lors des optimisations, un compromis doit être fait sur l'amplitude afin de faire correspondre le mieux possible l'intrados et l'extrados du profil sur le profil cible. Avec ces deux amplitudes par points clé, aucun compromis sur l'amplitude ne serait à faire. De plus, davantage de variables de design seraient disponibles, l'espace de design serait plus grand et donc de meilleurs résultats devraient être obtenus avec ce gain en flexibilité.

4. Pour la méthode de construction d'une B-Spline multi-contraintes en plusieurs endroits, il faudrait développer de nouvelles façons de générer le vecteur nodal, basées sur des grandeurs géométriques également. De plus, il faudrait que cette méthode donne également des courbes où les variations de la courbure sont douces sans avoir à ajuster manuellement les amplitudes afin de gagner en souplesse d'utilisation.
5. Enfin, les méthodologies présentées et ces améliorations devront être implémentées dans un environnement CAO. En effet, ces outils apportent plusieurs avantages, comme une visualisation intégrée, des bases de données, la possibilité d'une approche par ligne de commande et/ou en temps réel (par l'utilisation d'interfaces graphiques). De plus, la programmation de fonctions externes est de plus en plus aisée et un modèle a été développé pour le projet MOSAIC (tâche 5 - voir introduction).
6. Comme nous avons mentionné dans le dernier chapitre, des plans d'expériences et une sensibilité propre à chaque variable pourraient être utilisés afin de conduire de meilleures optimisations. Par exemple, faire les mêmes exercices, mais en éliminant certaines variables afin de réduire l'espace de design et comparer les résultats avec ceux obtenus afin d'illustrer les effets de la grandeur de l'espace de design.
7. Élaborer une structure permettant de procéder à des optimisations aérodynamiques pour constater quels effets ont les paramètres d'ingénierie sur la qualité et la rapidité des optimisations versus celles menées avec des points de contrôle B-

Splines comme variables de design. Observer l'évolution des performances aérodynamiques de l'aile en fonction de l'ajout de couches de contraintes.

8. Aussi, de nouvelles stratégies d'optimisation peuvent être adoptées afin de voir si elles seraient plus performantes dans le présent cas. L'utilisation de méthodes globales permettraient également d'explorer davantage l'espace de design.

BIBLIOGRAPHIE

- BENTAMY, A. 2003. *Modélisation Géométrique par NURBS pour le Design Aérodynamique des Ailes d'Avions*. Thèse de doctorat, École Polytechnique de Montréal.
- BERTIN, J. J. ET SMITH, M. L. 1998. *Aerodynamics for engineers*, Upper Saddle River, Prentice Hall, New Jersey. 668.
- BLOOR, M. I. G. ET WILSON, M. J. 1995. Efficient Parametrization for Generic Aircraft Geometry. *Journal of Aircraft*, 32, 1269–1275.
- CAMPBELL, R. L. 1992. An Approach to Constrained Aerodynamic Design with Application to Airfoils. Technical Report TP-3260, NASA.
- CHANG, C., TORRES, F. J. ET TUNG, C. 1995. Geometrical Analysis of Wing Sections. Technical Memorandum 110346, NASA.
- GAUVIN, J. 1995. *Leçons de programmation mathématique*. Éditions de l'École Polytechnique de Montréal.
- GLAUERT, H. 1947. *The Element of Airfoil and Airscrew Theory*, Cambridge, England, chapitre The Transformation of a Circle Into an Aerofoil.
- HAIMES, R. 2000. *CAPRI : Computational Analysis Programming Interface*. M.I.T. A Solid Modelling Based Infra-structure for Engineering Analysis and Design.
- HARRIS, C. D. 1990. NASA Supercritical Airfoils. Technical Paper 2969, NASA.
- HICKS, R. M. ET HENNE, P. A. 1979. Wing Design by Numerical Simulation. AIAA Paper 79-0080 7, AIAA.
- JAMESON, A. 1994. *Optimum Design Methods in Aerodynamics*, Von Karman Institute for Fluids Dynamics, Belgique, chapitre Optimum Aerodynamic Design via Boundary Control. Lecture Series.

- LÉPINE, J. 1999. *Optimisation de la Représentation de Profils d'ailes pour Application au Design Aérodynamique*. Mémoire de maîtrise, École Polytechnique de Montréal.
- LÉPINE, J., TRÉPANIÉ, J.-Y. ET PÉPIN, F. 1999. ÇOptimization of Wing Profile Geometric Representation Using NURBSÈ. *Canadian Aeronautics and Space Institute*. Montréal, Canada.
- NEMEC, M., AFTOSMIS, M. J. ET PULLIAM, T. H. 2004. ÇCAD-Based Aerodynamic Design of Complex Configurations Using a Cartesian MethodÈ. *42nd AIAA Aerospace sciences Meeting*. Reno, Nevada.
- OYAMA, A. 2000. *Wing Design Using Evolutionary Algorithms*. Thèse de doctorat, Tohoku University.
- PIEGL, L. ET TILLER, W. 1995. *The NURBS Book*. Springer-Verlag.
- ROBINSON, G. M. ET KEANE, A. J. 2001. ÇConcise Orthogonal Representation of Supercritical AirfoilsÈ. *Journal of Aircraft*, 38.
- SAMAREH, J. A. 1999a. ÇA Novel Shape Parametrization ApproachÈ. Rapport technique TM-99-209116, NASA.
- SAMAREH, J. A. 1999b. ÇStatus and Future of Geometry Modeling and Grid Generation for Design and OptimizationÈ. *journal of aircraft*, 36, 97–104.
- SAMAREH, J. A. 1999c. ÇA Survey of Shape Parameterization TechniquesÈ. *CEAS/AIAA/ICASE/NASA Langley International Forum on Aeroelasticity and Structural Dynamics*. Williamsburg, VA.
- SMITH, R. E., BLOOR, M. I. G., WILSON, M. J. ET THOMAS, A. T. 1995. ÇRapid Airplane Parameteric Input Design (RAPID)È. *12th AIAA Computational Fluid Dynamics Conference*. San Diego.

SOBIECZKY, H. 1998. *Notes on Numerical Fluid Mechanics*, Vieweg-Verlag, vol. 68, chapitre Parametric Airfoils and Wings.

SUMMERFILED, M. 1982. *Transonic Aerodynamics*. AIAA.

THEODORSEN, T. ET GARRICK, I. E. 1934. ÇGeneral Potential Theory of Arbitrary Wing SectionsÈ. Rapport technique Report no 452, NACA.

VANDERPLAATS, G. N. 1999. *Numerical Optimization Techniques for Engineering Design*. Vanderplaats Research & Development, Inc.

ANNEXE I

RAPPEL : COURBES B-SPLINAIRES

CETTE annexe est consacrée à l'étude des courbes B-Splines. Elle fait un rappel de la définition mathématique de ces courbes, des différentes propriétés de celle-ci ainsi que quelques algorithmes de manipulations de bases. La référence complète à ce sujet est le livre *The NURBS Book* de Piegl et Tiller (1995). Comme nous l'avons vu à l'équation 1.23 de la section 1.1.7, une courbe B-Spline est décrite par l'équation :

$$\mathbf{C}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i \quad (\text{I.1})$$

Où les \mathbf{P}_i sont les $n+1$ points de contrôle et les $N_{i,p}(u)$ les fonctions de bases B-Splinaires. La première partie de ce rappel a pour sujet ces fonctions de bases.

I.1 Fonctions de bases

Les fonctions de bases sont des fonctions polynomiales par morceau générées à partir du vecteur nodal U associé à la courbe $C(u)$ et du degré p de cette dernière. Le vecteur nodal est une suite monotone croissante de scalaire définissant l'intervalle paramétrique de la courbe. Il est constitué de m éléments, appelés nœuds, généralement noté, $U = u_0, u_1, \dots, u_i, \dots, u_{m-1}, u_m$. Le nombre de nœuds, de points de contrôle et le degré sont reliés par la relation $m = n + p + 1$. Dans le cas présent, nous nous attarderons seulement

aux vecteurs nodaux dit ouverts, ou non-périodiques. Ceux-ci ont la forme :

$$U = \{ u_0, u_1, \dots, u_p, u_{p+1}, u_{p+2}, \dots, u_{m-p-2}, u_{m-p-1}, u_{m-p}, \dots, u_{m-1}, u_m \}$$

$$U = \{ \underbrace{a, \dots, a}_{p+1}, u_{p+1}, u_{p+2}, \dots, u_{m-p-2}, u_{m-p-1}, \underbrace{b, \dots, b}_{p+1} \} \quad (I.2)$$

Dans la notation des fonctions de bases $N_{i,p}(u)$, i est le numéro de la fonction de base et p son degré. Ainsi, $N_{3,2}(u)$ est la troisième fonction de base de degré second. Les fonctions de bases se définissent récursivement de la façon suivante :

$$N_{i,0} = \begin{cases} 1 & \text{Si } u_i \leq u < u_{i+1} \\ 0 & \text{Autrement} \end{cases} \quad (I.3)$$

$$N_{i,p}(u) = \frac{u - u_i}{u_{i+p} - u_i} N_{i,p-1}(u) + \frac{u_{i+p+1} - u}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u)$$

La dérivée des fonctions de bases s'exprime aussi de manière récursive :

$$N'_{i,p}(u) = \frac{p}{u_{i+p} - u_i} N_{i,p-1}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (I.4)$$

Voyons maintenant les propriétés de ces fonctions de bases.

Non-négativité : Toutes les fonctions de bases, en tous point à l'intérieur de l'intervalle paramétrique sont plus grandes ou égales à zéro, $N_{i,p}(u) \geq 0$.

Partition de l'unité : En tous point à l'intérieur de l'intervalle paramétrique, la sommes des fonctions de bases donne toujours l'unité, $\sum_{i=0}^n N_{i,p}(u) = 1$, *allu*

Maximum unique : À l'exception des fonction de bases de degré $p = 0$, toutes les fonctions de bases atteignent une valeur maximum unique dans l'intervalle paramétrique

Support compact : Les fonctions de bases sont nulles à l'extérieur de leur intervalle

d'influence, $N_{i,p}(u) = 0$ si $u < u_i$ ou $u \geq u_{i+p+1}$. Le corollaire de cette propriété est que pour tout intervalle, il y a au plus $p + 1$ fonctions de bases non-nulles. Ce sont les fonctions $N_{i-p,p}$ à $N_{i,p}$.

Dérivées : Les fonctions de bases sont indéfiniment différentiable à l'intérieur de tout intervalle. Cependant, aux nœuds, elles ne sont que différentiable $p - k$ fois, où k est la multiplicité du nœud.

Interpolation aux extrémités : En ayant un vecteur nodal ouvert, la multiplicité des nœuds extrêmes engendre que $N_{0,p}(a) = 1$ et $N_{n,p}(b) = 1$. Autrement dit, la courbe passe toujours par le premier et dernier point de contrôle.

I.2 Courbes B-Splines

À la lumière des propriétés des fonctions de bases, nous pouvons maintenant caractériser davantage les courbes B-Spline. Ce sont des courbes polynomiales par morceaux. Il y a toujours autant de fonctions de bases que de points de contrôle, de sorte que chacune d'entre elle agit comme un facteur pondérant son point de contrôle associé en fonction de l'avancement paramétrique u . Or, nous savons que les fonctions de bases sont non-nulles à l'intérieur de leur intervalle. Donc, en tous points paramétrique, il y a un maximum de $p + 1$ points de contrôle qui influencent la position de la courbe. Cette dernière est donc le barycentre des points de contrôle actifs. Ceci est appelé la propriété de l'enveloppe convexe : la courbe est toujours contenue dans l'enveloppe convexe de ses points de contrôle.

Une autre caractéristique intéressante découlant du fait que les fonctions de bases sont des polynômes par morceaux, est le contrôle local sur la courbe. En effet, en modifiant la position d'un point de contrôle \mathbf{P}_i , la courbe n'est modifiée que dans l'intervalle $u \in [u_i, u_{i+p+1}[$.

I.3 Dérivées des courbes B-Splines

Comme le font remarquer Piegl et Tiller (1995), la dérivée peut être exprimée de deux façons. Tout d'abord, comme étant la dérivée des fonctions de bases multipliée par les points de contrôle :

$$\mathbf{C}'(u) = \sum_{i=0}^n N'_{i,p}(u) \mathbf{P}_i \quad (\text{I.5})$$

La deuxième façon, exprime la dérivée comme la multiplication des fonctions de bases de degré $p-1$ par de nouveaux points de contrôle, les \mathbf{Q}_i . À partir de l'équation I.5, nous allons trouver cette dernière façon d'exprimer la dérivée. Développons tout d'abord la dérivée des fonctions de bases :

$$N'_{i,p} = \frac{p}{u_{i+p} - u_i} N_{i,p-1}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \quad (\text{I.6})$$

En substituant l'équation I.6 dans I.5, nous obtenons :

$$\begin{aligned} \mathbf{C}'(u) &= \sum_{i=0}^n \left(\frac{p}{u_{i+p} - u_i} N_{i,p-1}(u) - \frac{p}{u_{i+p+1} - u_{i+1}} N_{i+1,p-1}(u) \right) \mathbf{P}_i \\ &= \left(p \sum_{i=-1}^{n-1} N_{i+1,p-1}(u) \frac{\mathbf{P}_{i+1}}{u_{i+p+1} - u_{i+1}} \right) - \left(p \sum_{i=0}^n N_{i+1,p-1}(u) \frac{\mathbf{P}_i}{u_{i+p+1} - u_{i+1}} \right) \quad (\text{I.7}) \\ &= p \frac{N_{0,p-1}(u) \mathbf{P}_0}{u_p - u_0} - p \sum_{i=0}^{n-1} N_{i+1,p-1}(u) \frac{\mathbf{P}_{i+1} - \mathbf{P}_i}{u_{i+p+1} - u_{i+1}} - p \frac{N_{n+1,p-1}(u) \mathbf{P}_n}{u_{n+p+1} - u_{n+1}} \end{aligned}$$

Dans la représentation B-Spline, les termes comprenant une division par zéro doivent être éliminés pour conserver la cohérence. Les premier et dernier termes de l'équation I.7 sont justement divisés par zéro, nous devons donc les éliminer. Nous pouvons donc

exprimer la dérivée première de la façon suivante :

$$\mathbf{C}'(u) = \sum_{i=0}^{n-1} N_{i+1,p-1}(u) \mathbf{Q}_i \quad (\text{I.8})$$

Avec

$$\mathbf{Q}_i = p \frac{\mathbf{P}_{i+1} - \mathbf{P}_i}{u_p - u_0} \quad (\text{I.9})$$

En actualisant le vecteur nodal, nous pouvons exprimer l'équation I.8 de la façon suivante (changement sur le i des fonction de bases) :

$$\mathbf{C}'(u) = \sum_{i=0}^{n-1} N_{i,p-1}(u) \mathbf{Q}_i \quad (\text{I.10})$$

Le nouveau vecteur nodal U' s'obtient en éliminant les premier et dernier éléments de U (u_0 et u_m) et nous obtenons :

$$U' = \{ \underbrace{0, \dots, 0}_p, u_p, \dots, u_{m-p-2}, \underbrace{1, \dots, 1}_p \} \quad (\text{I.11})$$

La relation entre les nouveaux nœuds internes par rapport aux anciens est $u'_i = u_{i+1}$. La courbe représentant la dérivée, n'est rien d'autre qu'une nouvelle B-Spline de degré $p - 1$ qui utilise de nouveaux points de contrôle, dérivés des originaux. Nous pouvons généraliser la formulation des équations I.9 et I.10 pour n'importe quelle ordre de dérivée $k < p$.

$$\mathbf{C}^{(k)}(u) = \sum_{i=0}^{n-k} N_{i,p-k}(u) \mathbf{P}_i^{(k)}$$

Avec

$$\mathbf{P}_i^{(k)} = \begin{cases} \mathbf{P}_i & \text{si } k = 0 \\ \frac{p - k + 1}{u_{i+p+1} - u_{i+k}} (\mathbf{P}_{i+1}^{(k)} - \mathbf{P}_i^{(k)}) & \text{si } k > 0 \end{cases} \quad (\text{I.12})$$

I.4 Manipulations contraintes de courbes B-Splines

L'objectif ici est de contraindre la courbe et/ou ses dérivées à une valeur déterminée à un certain avancement u . Rappelons encore l'équation générale des B-Splines (eq. 1.23) :

$$\mathbf{C}(u) = \sum_{i=0}^n N_{i,p}(u) \mathbf{P}_i \quad (\text{I.13})$$

La $k^{\text{ième}}$ de dérivée cette expression est donné par :

$$\mathbf{C}^{(k)}(u) = \sum_{i=0}^n N_{i,p}^{(k)}(u) \mathbf{P}_i \quad (\text{I.14})$$

Exprimé ainsi, la dérivée d'ordre 0 est la courbe originale. Cette notation permet de ne parler qu'en terme de dérivée, même si le sujet est la courbe elle-même. Cette expression définit les dérivées d'une B-Spline comme étant de nouvelles B-Splines. Ainsi, une courbe représentant une dérivée d'ordre k est une multiplication des mêmes points de contrôle par la $k^{\text{ième}}$ dérivée des fonctions de bases.

L'objectif ici est de contraindre une courbe à certains points paramétriques de sorte que celle-ci maintienne ou atteigne certaines propriétés. Il est souvent plus pratique d'exprimer les changements des $k^{\text{ième}}$ dérivées plutôt que des dérivées elles mêmes. De cette façon, si nous devons conserver certaines propriétés, nous n'avons qu'à imposer des changements nuls. Nous exprimons le changement des dérivées à un point paramétrique u_r comme suit :

$$\mathbf{C}^{(k)}(u_r) + \Delta \mathbf{C}^{(k)}(u_r) = \sum_{i=0}^n N_{i,p}^{(k)}(u_r) (\mathbf{P}_i + \Delta \mathbf{P}_i) \quad (\text{I.15})$$

De l'expression I.15, nous pouvons extraire le déplacement des points de contrôle $\Delta \mathbf{P}_i$

en fonction du changement de la $k^{\text{ième}}$ dérivée pour un point paramétrique u_r donné :

$$\Delta \mathbf{C}^{(k)}(u_r) = \sum_{i=0}^n N_{i,p}^{(k)}(u_r)(\Delta \mathbf{P}_i) \quad (\text{I.16})$$

En développant cette expression et en la mettant sous forme vectorielle, elle se simplifie à :

$$\Delta \mathbf{C}^{(k)}(u_r) = \mathbf{B}^{(k)}(u_r) \Delta \mathbf{P}^T \quad (\text{I.17})$$

où $\mathbf{B}^{(k)}(u_r)$ est la $k^{\text{ième}}$ dérivée du vecteur des fonctions de bases $N_{0,p}(u_r), \dots, N_{n,p}(u_r)$.

L'équation I.17 est linéaire avec $n + 1$ inconnus, les $\Delta \mathbf{P}_i$. Mais, due à la propriété de support local des B-Splines (voir annexe I), il n'y a en fait que $p + 1$ inconnues, les autres $N_{i,p}(u_r)$ étant par construction nulle. Ceci s'explique par le fait que ces points de contrôle, n'étant pas dans la région d'influence du point paramétrique u_r , n'ont aucun effet sur la courbe. Les déplacements qui leurs sont associés $\Delta \mathbf{P}_i$ sont donc nuls également.

Il est possible d'appliquer simultanément plusieurs de ces contraintes sur une courbe. L'équation I.17 doit être solutionnée pour chacune des contraintes, donnant naissance à un système d'équation avec au plus $n + 1$ inconnues, les $\Delta \mathbf{P}_i$. Ce système s'exprime comme suit :

$$\Delta \mathbf{C}_r^{(k)} = [\mathbf{B}] \Delta \mathbf{P}^T \quad (\text{I.18})$$

Où $[\mathbf{B}]$ est la matrice du système à résoudre, contenant sur ces $n + 1$ lignes, le vecteur des fonctions de bases et ses dérivées évalué aux différents u_r . Il faut noter que dans cette équation, les contraintes doivent être listées d'une façon quelconque. L'ordre dans cette liste peut être arbitraire, mais il est logique de les trier d'abord selon u_r , ensuite selon k .

Aussi, comme mentionné plus haut, seulement un sous-groupe des points de contrôle \mathbf{P}_i ont une influence à un u_r donné. Ce qui fait que plusieurs colonne de la matrice \mathbf{B} peuvent égaier zéro. L'algorithme de résolution devra prendre en compte ce fait pour

monter un système qui peut contenir moins de $n + 1$ inconnues. Notons N ce nombre d'inconnues.

Il est donc possible que le système soit sous-, pleinement- ou sur-déterminé. La méthode de résolution ne supporte que les cas sous- ou pleinement-déterminé avec toutes les rangées de B linéairement indépendantes. Le cas d'un système sur-déterminé ($N > M$) est obtenu si plus de contraintes que la courbe ne peut le supporter sont appliquées simultanément. On ne peut pas mettre plus de $p + 1$ contraintes dans chaque intervalle nodale, car ceux-ci ne sont sous l'influence que de $p + 1$ points de contrôle. Si on ne respecte pas cette condition, un système sur-déterminé est obtenu. Dans cette éventualité, les algorithmes d'insertion nodale ou d'élévation du degré peuvent être appliqués pour retrouver un système sous- ou pleinement-déterminé. Nous allons maintenant voir la méthode de résolution.

Solution au déplacement minimum

Dans le cas d'un système pleinement déterminé, la solution est obtenue directement par inversion de la matrice $[B]$. Pour un système sous-déterminé, le processus de résolution est plus complexe, étant donné l'infinité de solutions possibles. Dans ce cas, comme le propose Piegls et Tiller (1995), nous voulons celle qui va minimiser le déplacement global des points de contrôle, i.e. celle qui a la norme $\|\Delta P_i\|$ la plus petite. Nous voulons résoudre le système sous déterminé :

$$[B]\Delta P = \Delta C \quad (I.19)$$

Nous pouvons exprimer ΔP comme :

$$\Delta P = [B]^T \lambda + z \quad (I.20)$$

Il est alors possible de développer le carré de la norme comme suit :

$$\begin{aligned}
 |\Delta \mathbf{P}|^2 &= ([\mathbf{B}]^T \lambda + \mathbf{z})^T \cdot ([\mathbf{B}]^T \lambda + \mathbf{z}) \\
 &= ([\mathbf{B}]^T \lambda)^T \cdot ([\mathbf{B}]^T \lambda) + \mathbf{z}^T \cdot \mathbf{z} \\
 &= |[\mathbf{B}]^T \lambda|^2 + |\mathbf{z}|^2
 \end{aligned} \tag{I.21}$$

En incluant maintenant la définition des contraintes (eq. I.19) et en substituant $\Delta \mathbf{P}$ tel que le définit l'équation I.20 :

$$\begin{aligned}
 \Delta \mathbf{C} &= [\mathbf{B}] \Delta \mathbf{P} \\
 &= [\mathbf{B}] ([\mathbf{B}]^T \lambda + \mathbf{z}) \\
 &= [\mathbf{B}] [\mathbf{B}]^T \lambda + [\mathbf{B}] \mathbf{z} \\
 &= [\mathbf{B}] [\mathbf{B}]^T \lambda
 \end{aligned} \tag{I.22}$$

En solutionnant I.22 pour λ :

$$\lambda = ([\mathbf{B}] [\mathbf{B}]^T)^{-1} \Delta \mathbf{C} \tag{I.23}$$

Les composants de \mathbf{z} dans l'équation I.21 ne sont pas significatives pour le système, mais contribuent tout de même à la longueur de la norme de $\Delta \mathbf{P}$. Nous trouvons la solution minimisant le déplacement en fixant $\mathbf{z} = 0$. Nous obtenons donc :

$$\Delta \mathbf{P} = [\mathbf{B}]^T ([\mathbf{B}] [\mathbf{B}]^T)^{-1} \Delta \mathbf{C} \tag{I.24}$$

En résolvant I.24, nous obtenons le déplacement que doivent subir certains points de contrôle. En appliquant ces translations, nous obtenons une courbe respectant les contraintes imposées.

La matrice $([\mathbf{B}] [\mathbf{B}]^T)$ n'est pas singulière si les lignes de $[\mathbf{B}]$ sont linéairement indépendantes. La condition suffisante est de vérifier que $\det([\mathbf{B}] [\mathbf{B}]^T) \neq 0$. C'est de cette

méthode qu'est inspirée la méthodologie présentée à la section 3.1.2. La différence étant qu'il n'y a pas de courbe sur laquelle s'appuyer, et donc, les $\Delta\mathbf{C}$ sont les positions des points de contrôle directement, les \mathbf{P}_i .

ANNEXE II

RAPPEL : MÉTHODES D'OPTIMISATION À GRADIENT QUASI-NEWTON BFGS

DANS les dernières années, une plus vaste et efficace utilisation de l'optimisation est observée dans une multitude de domaines. Plusieurs approches d'optimisation sont répertoriées aujourd'hui. Les plus populaires sont les méthodes à gradient et les algorithmes génétiques (AG). Notons aussi les méthodes à recuit simulé et les recherches de motifs (*pattern search*) comme autres alternatives. Dans le domaine général de l'optimisation, chacune de ces méthodes ont leurs propres champs de recherche associés afin d'en continuer le développement.

II.1 Formulation

Les méthodes d'optimisation se divisent en deux grandes catégories. Les méthodes à gradient, qui donnent une solution locale, alors que les méthodes heuristiques, tels que les AG, cherchent une solution plus globale. Cependant, dans ce dernier cas, les temps de calcul sont plus importants. Les méthodes à gradient doivent démarrer leur recherche à partir d'une solution initiale, contrairement aux AG qui s'initialisent avec une population. Les premières se basent sur une direction de descente pour améliorer la fonction objectif, alors que la seconde utilise le concept de sélection naturelle pour produire de meilleurs individus.

Le recours aux méthodes d'optimisations permet d'améliorer le design d'un système ou d'une pièce. Son principal atout est d'aider l'ingénieur dans son travail de design pour

lui permettre de trouver le résultat optimal, et idéalement, de lui libérer du temps afin qu'il puisse consacrer plus de temps à la création. La formulation mathématique d'un problème d'optimisation avec contraintes est :

$$\begin{aligned}
 & \min_{\mathbf{x}} f(\mathbf{x}) \\
 & \text{tel que : } \quad g_i(\mathbf{x}) \geq 0 \quad 0 \leq i < l \\
 & \quad \quad \quad h_i(\mathbf{x}) = 0 \quad 0 \leq i < m \\
 & \quad \quad \quad x_i^L \leq x_i \leq x_i^U \quad 0 \leq i < n
 \end{aligned} \tag{II.1}$$

où $f(\mathbf{x})$ est la fonction coût ou objectif, \mathbf{x} est le vecteur de design comprenant n variables, les $g_i(\mathbf{x})$ sont les l contraintes d'inégalité, les $h_i(\mathbf{x})$ sont les m contraintes d'égalité et enfin, les x_i^L et x_i^U sont les bornes inférieures et supérieures des variables de design.

Dans le cadre de cette maîtrise, seule la méthode à gradient quasi-Newton BFGS a été utilisée. La facilité de mise en œuvre, la disponibilité du code, l'aspect général de la méthode et les bons résultats obtenus ont justifié son utilisation.

II.2 Optimiseur à base de gradient quasi-Newton BFGS

Parmi les méthodes à gradient, les plus répandues sont celles qui utilisent une approximation du Hessien pour représenter la variation de la fonction coût. Celles-ci sont appelées les méthodes quasi-Newton. Elles se servent de l'information relative au gradient afin d'améliorer la fonction objectif. Le problème de minimisation sans contraintes s'expose comme suit :

$$\min_{\mathbf{x}} f(\mathbf{x}) \tag{II.2}$$

Le développement de Taylor de second ordre de $f(\mathbf{x})$ autour de $f(\mathbf{x}_0)$ est :

$$\begin{aligned}
 f(\mathbf{x}) &= f(\mathbf{x}_0) + \sum_{i=1}^n \frac{\partial f(\mathbf{x}_0)}{\partial x_i} (x_i - x_{0i}) + \\
 &\quad \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \frac{\partial^2 f(\mathbf{x})}{\partial x_i \partial x_j} (x_i - x_{0i}) (x_j - x_{0j}) + O^{(2)}(\mathbf{x} - \mathbf{x}_0) \\
 &= f(\mathbf{x}_0) + \nabla f(\mathbf{x}_0) (\mathbf{x} - \mathbf{x}_0) + \frac{1}{2} (\mathbf{x} - \mathbf{x}_0)^T [\mathbf{H}](\mathbf{x}_0) (\mathbf{x} - \mathbf{x}_0) + O^{(2)}
 \end{aligned} \tag{II.3}$$

où ∇ est l'opérateur gradient et $[\mathbf{H}]$ la matrice hessienne. À partir de la dernière équation, on peut déduire les conditions d'optimalité. On sait qu'un minimum se produit à un endroit où les pentes sont nulles et les dérivées secondes positives. La première situation, dite condition nécessaire d'optimalité, s'obtient lorsque le gradient est nul. La seconde, dite condition suffisante, se rencontre lorsque le Hessian est strictement défini positif. Un point respectant ces deux critères est dit optimal et est noté \mathbf{x}^* . Ces conditions s'expriment comme suit (Gauvin 1995) :

$$\begin{aligned}
 \nabla f(\mathbf{x}^*) &= 0 \\
 \det [\mathbf{H}(\mathbf{x}^*)] &> 0
 \end{aligned} \tag{II.4}$$

Pour trouver un tel point, les méthodes à gradient partent d'un point original \mathbf{x}_0 qui est amélioré par itérations successives. À chaque itération k , une recherche de ligne est exécutée dans une direction de descente \mathbf{d}_k , définie un peu plus loin. Dans cette direction, un pas d'avancement α_k est recherchée afin de diminuer au maximum la fonction coût. Les valeurs du vecteur de design \mathbf{x}_k doivent être actualisées en fonction de cette descente avec :

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{d}_k \tag{II.5}$$

Les méthodes de type Newton, en opposition à celle de type quasi-Newton, calculent $[\mathbf{H}]$ directement. Évaluer $[\mathbf{H}]$ directement, quand cela est possible, implique une grande

quantité de calculs numériques. Les méthodes quasi-Newton contournent ceci en construisant une approximation de $[\mathbf{H}]$ basée sur le comportement de $f(\mathbf{x})$ et de $\nabla f(\mathbf{x})$ au point de design courant et d'une technique de mise à jour appropriée.

Une grande quantité de méthode de mise à jour du Hessien a été développée. Cependant, la formulation de Broyden, Fletcher, Goldfarb, et Shanno (BFGS) (Vanderplaats 1999) est réputée comme étant la plus efficace pour un usage général. La formulation donnée par BFGS s'exprime comme suit :

$$[\mathbf{H}]_{k+1} = [\mathbf{H}]_k + \frac{\mathbf{q}_k \mathbf{q}_k^T}{\mathbf{q}_k^T \mathbf{s}_k} - \frac{[\mathbf{H}]_{k+1}^T \mathbf{s}_k^T \mathbf{s}_k [\mathbf{H}]_k}{\mathbf{s}_k^T [\mathbf{H}]_k \mathbf{s}_k} \quad (\text{II.6})$$

Où

$$\begin{aligned} \mathbf{s}_k &= \mathbf{x}_{k+1} - \mathbf{x}_k \\ \mathbf{q}_k &= \nabla f(\mathbf{x}_{k+1}) - \nabla f(\mathbf{x}_k) \end{aligned} \quad (\text{II.7})$$

Au début du calcul, n'importe quelle matrice définie positive symétrique peut faire l'affaire. Dans la méthode utilisée, on impose la matrice identité comme point de départ, $[\mathbf{H}]_0 = I$.

Le gradient est soit calculé analytiquement, soit obtenu via une méthode de différenciation numérique, comme les différences finies. Cette dernière option oblige à faire varier chacune des variables de design et à voir l'effet sur la fonction coût. Bien que cette technique induit un bruitage de la fonction coût, elle est souvent utilisée, car la fonction est rarement connue analytiquement.

Le Hessien est au cœur de cette recherche de ligne en fournissant la direction de descente. Exprimée mathématiquement, on a :

$$\mathbf{d}_k = -[\mathbf{H}]_k^{-1} \cdot \nabla f(\mathbf{x}_k) \quad (\text{II.8})$$

Dans cette direction, il faut trouver le pas d'avancement qui réduira le plus possible la fonction objectif. Pour ce faire, un scalaire α multipliant le vecteur de descente est nécessaire. Une plage de variation est déterminée et ensuite, la meilleure valeur α_k est trouvée. À cette valeur les critères suivants doivent être satisfaits :

$$\begin{aligned} f(\mathbf{x}_k + \alpha_k \cdot \mathbf{d}_k) &\leq f(\mathbf{x}_k) + c_1 \alpha \nabla f(\mathbf{x}_k)^T \mathbf{d}_k \\ \nabla f(\mathbf{x}_k + \alpha_k \cdot \mathbf{d}_k)^T \mathbf{d}_k &\geq c_2 \alpha \nabla f(\mathbf{x}_k)^T \mathbf{d}_k \end{aligned} \quad (\text{II.9})$$

Où c_1 et c_2 sont des constantes et $0 < c_1 < c_2 < 1$. Le premier critère assure une descente suffisante, alors que le second contraint le pas d'avancement à ne pas être trop petit. La valeur typique de α pour initialiser la recherche est l'unité. Une bonne méthode pour trouver le bon α , est de le diviser par 2 à chaque fois que le critère de descente suffisante n'est pas respecté. Un critère sur le nombre maximal d'itérations s'avère utile pour limiter les temps de calcul ou éviter de suivre trop longtemps inutilement des directions asymptotiques. Pour cette maîtrise, la méthode d'optimisation quasi-Newton BFGS est tirée du travail de Lépine (1999), dans lequel ce dernier a implémenté la méthode décrite dans cette section.

ANNEXE III

DESCRIPTION DES FICHIERS D'ENTRÉES

CETTE annexe est consacrée à la description des fichiers d'entrées requis pour la construction de l'aile. Les informations relatives à l'aile sont regroupées selon leur nature dans des fichiers distincts. Il en résulte que trois fichiers sont nécessaires ; un premier pour les paramètres généraux de construction de l'aile, un second pour les informations relatives à la forme en plan de l'aile et enfin un troisième pour les informations géométriques des profils guides.

Le module lit dans un premier temps un fichier d'entrée, dans lequel se trouvent tous les noms de fichiers et chemins nécessaires pour son exécution. Ce fichier contient les trois noms de fichiers pour les paramètres géométriques de l'aile aux stations de définitions, pour la forme en plan et pour les informations sur sa construction. Les chemins définissent l'endroit où les données se trouvent (les trois fichiers précédemment cités) et où les résultats seront écrits.

III.1 Paramètres généraux de construction de l'aile

Ce fichier est constitué d'information de nature générale, comme les couches actives ou les nombre de points de discrétisation désiré. Les informations contenues dans ce fichier ne peuvent pas être des variables de design lors d'optimisations. Des lignes de commentaires peuvent être ajoutées dans un fichier. Le symbole "#" signifie que la ligne n'est pas lue, tout comme "//" indique que le reste de la ligne est un commentaire. Un fichier de ce type est représenté au tableau III.1.

TAB. III.1: Nature du fichier de construction

Nom de la borne	Description
#Éléments de construction (0 pour non, 1 pour oui)	
Struct 1	// Structure
BecVolet 1	// Bec et Volet
Avance 1	// Avancer
SSpar 0	// Longerons auxiliaires dans la partie inboard
SlatDroit B	// Variation des composantes 'z' des lignes de
FlapDroit 0	// recouvrement linéaire
	// R,B,T=Section oublier
#Numéro de la méthode de construction des splines	
MethSpline 1	//1 Collage de B-Splines contraintes aux extrémités
	//2 Avec optimisation des amplitudes
	//3 B-Spline multi-contrainte
	//4 Avec optimisation des amplitudes
#Numéro de la méthode de construction du vecteur nodal	
VectNod 5	//Génération du vecteur nodal par la méthode
	// 0 -> de moyenne des courbures
	// 1 -> poids dérivée-courbure
	// 2 -> tier-tier modifiée
	// 3 -> uniforme
	// 5 -> moyenne améliorée des courbures
#Paramètre de discrétisation de l'aile	
NbStations 4	// Nombre de sections voulus
StationAutomatique 10 45 70 110	// si non présent, station
	// déterminée de façon automatique, si présent,
	// la liste de nombre doit correspondre à
	// l'emplacement des stations
#Paramètre de discrétisation des profils	
NbTot 100	// Nombre de points de discrétisation au total
NbEx 100	// Nombre de points de discrétisation sur l'extrados
NbIn 100	// Nombre de points de discrétisation sur l'intrados
NbFE 30	// Nombre de points de discrétisation pour le volet
NbFI 30	// sur l'extrados et l'intrados
NbE 100	// Nombre de points de discrétisation pour la partie
NbI 90	// interne de l'extrados et de l'intrados
NbSE 20	// Nombre de points de discrétisation pour le bec
NbSI 30	// sur l'extrados et l'intrados
Decal 100.	// Paramètre de décalage sur la courbure pour la
	// discrétisation

Les différentes bornes ont toutes leurs caractéristiques. Voici ce que contrôle chacune d'entre elles :

Struct, BecVolet, Avance et SSpar : Interrupteurs des couches des paramètres cités à la section 4.2.1. Chaque couche active quelques points clés sur les profils guides. Si une couche est active, le programme lira les informations correspondantes dans le fichier de paramètres géométriques en s'en servant pour la génération de l'aile.

SlatDroit et FlapDroit : Paramètres contrôlant la surface des bec et volet. Ils permettent d'imposer une surface réglée sur toute l'envergure comme le bord d'attaque et une surface réglée cassée comme bord de fuite. Autrement dit, les variations d'élévation sont constantes sur toute l'envergure, $\frac{\partial}{\partial z} = 0$. Pour ce faire, les informations relatives à un profil guide sont négligées pour construire la surface hypersustentatrice. La surface est générée par interpolation entre les deux profils actifs et extrapolation sur le troisième. Si le sélecteur indique "0" les surfaces sont générées normalement. S'il indique "R", la surface n'interpolera pas le profil à l'emplanture, s'il indique "B", le profil de la cassure ne sera pas considéré et s'il indique "T", ce sera le profil du bout.

MethSpline : Indique le numéro de la méthode d'interpolation avancée à utiliser. La méthode #1 est celle de la section 3.1.1, la méthode #2 fait référence à la section 3.1.2 et enfin la méthode #3 est comme la #2, mais en optimisant les amplitudes tel que décrit à la section 3.1.3. La méthode par défaut est la #1.

VectNod : Si la méthode de construction des splines est **MethSpline**= 1, nous avons le choix de la méthode de construction du vecteur nodal (voir la section 3.1.1.2). Par défaut, la moyennisation des courbures améliorées est utilisée.

NbStation : Correspond au nombre de stations voulues sur l'aile finale. Ce nombre n'intervient pas sur la géométrie, mais bien sur la discrétisation de celle-ci.

StationAutomatique : Lorsque cette borne est présente, elle doit être suivie d'une suite de **NbStation** entiers. Ces nombres correspondent à la position des stations

Y sur lesquelles les profils seront décrits. Ces nombres doivent être dans la plage de l'envergure, autrement un message d'erreur est renvoyé. Si la borne n'est pas présente, le système sélectionne lui-même les stations de discrétisation.

NbTot : Correspond au nombre de points de discrétisation par profil. Ce nombre, comme les suivant, n'intervient pas sur la géométrie, mais bien sur la discrétisation de celle-ci.

NbEx et **NbIn** : Correspond aux nombres de points de discrétisation que l'on retrouve sur l'extrados et l'intrados respectivement.

NbFE, **NbFI**, **NbE**, **NbI**, **NbSE** et **NbSI** : Correspond aux nombres de points de discrétisation sur les six sous-parties du profil : bec, volet et partie interne sur l'extrados et l'intrados.

Decal : Paramètre de décalage sur la courbure utilisé pour la discrétisation tel que décrit à la section 4.2.5.

III.2 Paramètres de la forme en plan de l'aile

Toutes les informations relatives à la forme en plan sont consignées dans ce fichier. On y retrouve la définition de l'enveloppe extérieure, la position relative des longerons ainsi que celle des lignes de recouvrement et la distribution de torsion de l'aile. Le tableau III.2 illustre un tel fichier.

L'enveloppe est définie par les trois cordes, la demi-envergure, la position de la cassure et l'angle de flèche. La station de l'implanture correspond au décalage par rapport à l'origine désiré. Les autres valeurs sont des pourcentages qui correspondent à la position sur la corde de la grandeur en commentaire. Ainsi, dans cet exemple, le longeron avant se situe à 15% sur le profil de la cassure. Les valeurs à l'implanture sont extrapolées suivant une droite. Notons que les cordes des profils guides données ici sont les cordes

TAB. III.2: Nature du fichier des paramètres de la forme en plan

Nom de la borne	Description
#Paramètres pour la forme en plan	
40.0	// Corde à l'emplanture
25.0	// Corde à la cassure
15.0	// Corde au bout
100.0	// Demi-envergure
0.0	// Station de l'emplanture
35.0	// Station de la cassure
30.0	// Angle de flèche
15.00	// %C du longeron avant à la cassure
65.00	// %C du longeron arrière à la cassure
20.00	// %C du longeron avant au bout
55.00	// %C du longeron arrière au bout
13.00	// %C du pt recouvrement SE cassure
18.00	// %C du pt recouvrement SE bout
1.950	// %C du pt recouvrement SI cassure
2.60	// %C du pt recouvrement SI bout
85.30	// %C du pt recouvrement FE cassure
81.10	// %C du pt recouvrement FE bout
75.15	// %C du pt recouvrement FI cassure
68.05	// %C du pt recouvrement FI bout
#Twist	
2 1.5 -.5	// Twist, Torsion de l'aile au R B T
#Élévation des stations	
2.079970 0.351880 -0.289370	// Élévation du bord d'attaque au R B T
#Translation	
261.8180	// Translation en x de la forme en plan

projetées sur x . Les cordes réelles sont le quotient de cette corde par le cosinus de l'angle de torsion, $c = \frac{c_x}{\cos(\alpha)}$. Les élévations correspondent aux translations verticales à appliquer sur les profils guides. Si elles sont toutes nulles, le bord d'attaque se confond avec le plan de référence de l'aile, $z = 0$. La translation en x , comme son nom l'indique, permet de déplacer la forme en plan suivant x afin de permettre au bord d'attaque de la station 0 de prendre une valeur différente de zéro.

III.3 Paramètres géométriques de l'aile

Enfin, le troisième fichier renferme les caractéristiques géométriques de l'aile par le biais des trois profils guides. Les bornes sont donc suivies de trois chiffres, lesquels sont la valeur du paramètre à l'emplanture, à la cassure et au bout respectivement. Le fichier peut être plus ou moins long en fonction des couches activées dans le fichier de construction. L'entête, donnant l'information des points clés de la couche de base, doit quant à elle être toujours présente. Un fichier de paramètres géométriques est donné un tableau III.3.

Les caractéristiques géométriques sont données adimensionnellement. Pour les utiliser correctement, le programme les *redimensionne*, *i.e.* il multiplie leurs valeurs par la corde réelle du profil correspondant. Les angles sont additionnés de l'angle de torsion local. Seules les amplitudes, qui doivent demeurer sans dimension, demeurent inchangées. Voici une courte présentation des caractéristiques des bornes nécessaires pour la couche de base. Les autres bornes sont traitées séparément plus loin, car l'information est présentée différemment.

XX PointClé Y Y Y Y Y : L'information géométrique est regroupée par point clés.

La borne est le numéro unique du point clé (voir figure 4.4). À la suite de la borne, sur la même ligne se trouve le nom du point clé et une suite de cinq valeurs booléennes. Ces dernières indiquent quelle(s) contrainte(s) est(sont) active(s). On

TAB. III.3: Nature du fichier des paramètres géométriques des profils guides

Nom de la borne	Valeurs	Description
#		
#	# Paramètres géométriques à l'emplanture, à la cassure et au bout	
#		
#	# Paramètre de base	
#		
#	# num nom [PosX PosZ Pente Courbure Amplitude]switch	
#		
01	TrailingEdgeIntrados 0 1 1 0 0	
	-0.00183 -0.00257 -0.00255	// Pos. Z au bord de fuite intrados
	-1.3 -12.4 -12.4	// Pente au bord de fuite intrados
#		
02	MinimumPoint 1 1 0 1 0	
	0.3266 0.3425 0.3403	// Pos. X au point minimum
	-0.07767 -0.05441 -0.05439	// Pos. Z au point minimum
	0.3972 0.5662 0.5704	// Courbure au point minimum
#		
03	LeadingEdge 0 0 0 1 1	
	12.7951 178.56471 145.8726	// Courbure au bord d'attaque
	1 1 1	// Amplitude au bord d'attaque
#		
04	MaximumPoint 1 1 0 1 0	
	0.3239 0.4267 0.4295	// Pos. X au point maximum
	0.07163 0.06831 0.06837	// Pos. Z au point maximum
	0.3964 0.3082 0.3050	// Courbure au point maximum
#		
05	TrailingEdgeExtrados 0 1 1 0 0	
	-0.00183 -0.00257 -0.00255	// Pos. Z au bord de fuite extrados
	-13.0 -14.3 -14.3	// Pente au bord de fuite extrados
#		
Elev	1.0 0.5 -0.25	// Déplacement vertical (z) des profils guides
#		
#		
	⋮	

TAB. III.4: Nature du fichier des paramètres géométriques des profils guides (suite)

Nom de la borne	Valeurs	Description
:		
# Paramètres additionnels (si nécessaire)		
#		
# num nom [PosX PosZ Pente Courbure Amplitude]switch		
#		
11	RearSparIntrados 0 1 1 0 0	
	-0.051 -0.0317 -0.04125	// Pos. Z longeron arrière intrados
	6.239 9.5 8.25	// Pente longeron arrière intrados
#		
#		
12	FrontSparIntrados 0 1 1 0 0	
	-0.05 -0.05 -0.05	// Pos. Z longeron avant intrados
	-5. -5. -7.	// Pente longeron avant intrados
#		
13	FrontSparExtrados 0 1 1 0 0	
	0.055 0.055 0.0547	// Pos. Z longeron avant extrados
	6. 6. 6.5	// Pente longeron avant extrados
#		
14	RearSparExtrados 0 1 1 0 0	
	0.06 0.053 0.06	// Pos. Z longeron arrière extrados
	-3. -5.24618 -3.	// Pente longeron arrière extrados
#		
21	FlapIntrados 0 1 1 0 0	
	-0.0038 -0.015 -0.018	// Pos. Z jct. flap intrados
	5.5 7.5316 9.	// Pente jct. flap intrados
#		
22	SlatIntrados 0 1 1 0 0	
	-0.025 -0.020 -0.02	// Pos. Z jct slat intrados
	-25. -25. -15.641	// Pente jct slat intrados
#		

retrouve, dans l'ordre les interrupteurs de la position X , de la position Z , de l'angle de la pente \angle , de l'amplitude de la dérivée première \leftrightarrow et la courbure κ . Le nombre de lignes qui suivent la ligne d'identification est égal au nombre de contraintes actives. Sur ces lignes sont disposés la valeurs des paramètres pour les profils guides de l'implanture, de la cassure et du bout de l'aile.

Chaque couche à des traits distincts. Ainsi, sur la couche de base, on ne peut pas imposer la position X au bord de fuite, car elle est toujours de 1. Les pentes aux trois aux points clés de cette couche ne peuvent être contrainte, car elles sont imposées automatiquement par le système. De plus, sur les points clés des couche 1 (Structure) 2 (Bec & Volet) et 4 (Longeron auxiliaire), on ne peut pas spécifier la position X , car cette dernière provient de la forme en plan. De la même façon, la pente ne peut être imposée à la cavité, car elle est la même que l'angle d'incidence. Enfin, la courbure sur le point d'inflexion est naturellement toujours nulle.